



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA PODNIKATELSKÁ

FACULTY OF BUSINESS AND MANAGEMENT

ÚSTAV INFORMATIKY

INSTITUTE OF INFORMATICS

**NÁVRH, TVORBA A IMPLEMENTACE SOFTWARE
APLIKACE VE FIREMNÍM PROSTŘEDÍ**

DESIGN, CREATION AND IMPLEMENTATION OF SOFTWARE APPLICATIONS IN THE CORPORATE
ENVIRONMENT

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Juraj Zsiga

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Lukáš Novák, Ph.D.

BRNO 2021

Zadání diplomové práce

Ústav: Ústav informatiky
Student: **Bc. Juraj Zsiga**
Studijní program: Systémové inženýrství a informatika
Studijní obor: Informační management
Vedoucí práce: **Ing. Lukáš Novák, Ph.D.**
Akademický rok: 2020/21

Ředitel ústavu Vám v souladu se zákonem č. 111/1998 Sb., o vysokých školách ve znění pozdějších předpisů a se Studijním a zkušebním řádem VUT v Brně zadává diplomovou práci s názvem:

Návrh, tvorba a implementace softwarové aplikace ve firemním prostředí

Charakteristika problematiky úkolu:

Úvod
Vymezení problému a cíle práce
Teoretická východiska práce
Analýza problému a současné situace
Vlastní návrhy řešení, přínos návrhů řešení
Závěr
Seznam použité literatury
Přílohy

Cíle, kterých má být dosaženo:

Cílem práce je analyzovat, navrhnout a implementovat webovou aplikaci do firemního prostředí.

Základní literární prameny:

GÁLA, L., J. POUR a Z. ŠEDIVÁ. Podniková informatika. 2. přeprac. a aktualiz. vyd. Praha: Grada Publishing, 2009. ISBN 978-80-247-2615-1.

HARDCASTLE, E. Business Information Systems. Ventus Publishing ApS, 2008. ISBN 978-87-76-1-463-2.

PRETTYMAN, S. Learn PHP 7: object oriented modular programming using HTML5, CSS3, Javascript, XML, JSON, and MYSQL. Apress, 2015. ISBN 978-1-4842-1730-6.

SODOMKA, P. a H. KLČOVÁ. Informační systémy v podnikové praxi. 2. vyd. Brno: Computer Press, 2000. ISBN 978-80-251-2878-7.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2020/21

V Brně dne 28.2.2021

L. S.

Mgr. Veronika Novotná, Ph.D.
ředitel

doc. Ing. Vojtěch Bartoš, Ph.D.
děkan

Abstrakt

Diplomová práca sa zaoberá návrhom, tvorbou a implementáciou softvérovej aplikácie vo firemnom prostredí podniku Velká Pecka s.r.o., známejší pod názvom Rohlík. Prvým cieľom je analyzovať daný podnik a zistiť jeho nedostatky. Druhým, hlavným cieľom je práve tvorba a implementácia aplikácie, ktorá tieto nedostatky eliminuje. Samotné riešenie je mobilný softvér, ktorý prináša zamestnancom lepšie zaznamenávanie problémov v rámci skladových priestorov, čím sú ušetrené celkové prostriedky.

Abstract

The diploma thesis deals with the design, creation and implementation of a software application in the corporate environment of Velká Pecka s.r.o., better known as Rohlík. The first goal is to analyze the given company and find its shortcomings. The second, the main one, is to create and implement a software, which would eliminate them. The resulting solution is a mobile application, which improves issue reporting in their warehouses, thus saving resources overall.

Klíčové slová

Back-end, Databáza, Front-end, JavaScript, Mobilná aplikácia, React Native, Rohlík, Softvér, Vývoj

Keywords

Back-end, Database, Front-end, JavaScript, Mobile application, React Native, Rohlík, Software, Development

Citácia

ZSIGA, Juraj. *Návrh, tvorba a implementace softwarové aplikace ve firemním prostředí*. Brno, 2021. Dostupné z: <https://www.vutbr.cz/studenti/zav-prace/detail/133703>. Diplomová práce. Vysoké učení technické v Brně, Fakulta podnikatelská, Ústav informatiky. Vedoucí práce Lukáš Novák.

Prehlásenie

Prohlašuji, že předložená diplomová práce je původní a zpracoval jsem ji samostatně. Prohlašuji, že citace použitých pramenů je úplná, že jsem ve své práci neporušil autorská práva (ve smyslu Zákona č. 121/2000 Sb., o právu autorském a o právech souvisejících s právem autorským).

.....

Juraj Zsiga

15. mája 2021

Podakovanie

V prvom rade by som chcel poďakovať celej svojej rodine a svojim priateľom, bez ktorých by som to nikdy nezvládol, najmä svojim rodičom, ktorí ma vždy podporovali a boli mi vzorom počas doterajšieho štúdia. Taktiež chcem poďakovať svojmu vedúcemu diplomovej práce, Ing. Lukášovi Novákovi, Ph.D., za jeho podporu, ochotu a odborné rady, ktoré mi počas vypracovania vždy rád poskytol.

Obsah

Úvod	11
1 Vymedzenie problému a ciele práce	12
1.1 Vymedzenie problému	12
1.2 Ciele práce	12
2 Teoretické východiská práce	13
2.1 Základné termíny	14
2.1.1 Dáta	14
2.1.2 Programovací jazyk a kód	14
2.1.3 Programové vybavenie (softvér)	15
2.1.4 Softvérová aplikácia	16
2.1.5 Back-end	17
2.1.6 Databáza	17
2.1.7 Front-end	19
2.1.8 Framework	20
2.2 Pomocné nástroje	21
2.2.1 Figma	21
2.2.2 Expo	22
2.2.3 GitHub	22
2.3 Back-end technológie	23
2.3.1 JavaScript	23
2.3.2 Node.js	24
2.3.3 Express	25
2.3.4 GraphQL	25

2.3.5	MongoDB	25
2.4	Front-end technológie	27
2.4.1	React	27
2.4.2	React Native	27
2.5	Analytické nástroje	28
2.5.1	SLEPT	28
2.5.2	Porterova analýza	29
2.5.3	Rámec 7S	30
2.5.4	SWOT analýza	31
2.6	Projektové metódy	32
2.6.1	Lewinov model	32
2.6.2	Metóda RIPRAN	33
2.6.3	Metóda SCRUM a sprinty	35
3	Analýza problému a súčasnej situácie	36
3.1	Základné informácie	37
3.2	Predstavenie firmy	37
3.3	Ako funguje Rohlík.cz	38
3.4	Analýza vonkajších faktorov	39
3.4.1	Spoločenské faktory	39
3.4.2	Technologické faktory a ich vývoj	39
3.4.3	Ekonomické faktory	40
3.4.4	Politické faktory	40
3.4.5	Právne faktory	41
3.5	Porterova analýza	42
3.5.1	Hrozba súčasnej konkurencie	42
3.5.2	Hrozba nových firiem v odvetví	43
3.5.3	Hrozba rastúcej kompetencie dodávateľov	43
3.5.4	Hrozba rastúcej kompetencie zákazníkov	44
3.5.5	Hrozba substitučných produktov	44
3.6	Analýza vnútorných faktorov	45
3.7	Zhrnutie analýz - SWOT	49

4	Vlastné návrhy riešenia, prínos návrhov riešení	52
4.1	Aktuálny stav z hľadiska implementácie	52
4.2	Požiadavky na nový softvér	53
4.3	Predstavenie nového riešenia	54
4.4	Lewinov model	55
4.4.1	Fáza rozmrazenia	55
4.4.2	Fáza prechodu a aplikácia zmeny	58
4.4.3	Fáza zmrazenia	58
4.4.4	Záver po zavedení zmeny	58
4.5	Analýza rizík pomocou RIPRAN	59
4.5.1	Určenie sústavy	59
4.5.2	Identifikácia a ohodnotenie rizík	61
4.5.3	Výber opatrení a ich vplyv na výsledné hodnoty	62
4.6	Časový plán	63
4.6.1	Sprint 1	64
4.6.2	Sprint 2	65
4.6.3	Sprint 3	66
4.6.4	Sprint 4	67
4.7	Vlastný návrh softvérovej aplikácie	68
4.8	Porovnanie navrhnutého riešenia so súčasným	76
4.9	Analýza nákladov	78
4.10	Prínos nového riešenia	80
	Záver	81
	Literatúra	82

Úvod

V dnešnom svete je odvetvie informačných technológií veľmi dôležité, čo sa preukazuje aj v tomto období, kedy celý svet zastihla pandémia, kde sa sila informačných technológií preukázala najmä v oblasti komunikácie.

Jeden z hlavných nástrojov informatiky je samozrejme samotný návrh rôznych aplikácií a programov, ktoré slúžia na pomoc či už v súkromnom alebo profesionálnom živote. Jadrom tejto práce bude návrh a tvorba aplikácie s názvom Rohcket a jej samotná implementácia pre online obchod Rohlík. Cieľom práce je práve navrhnúť a implementovať daný software do reálneho chodu v ich firemnom prostredí. V tomto prípade by sa navrhované riešenie využilo v rámci skladových priestorov a manažmentu problémov.

Samozrejme, náplňou tejto práce nie je len samotná implementácia, ale aj analýza celkového návrhu z hľadiska projektového manažmentu ako takého.

V prvej kapitole diplomovej práce budú v krátkosti prezentované vymedzenie problému a ciele práce.

V ďalšej kapitole budú všeobecne opísané pojmy a nástroje, ktoré budú použité pri analýze firmy, projektovej analýze a samotnej implementácii mobilnej aplikácie.

Tretia kapitola má na starosti predstaviť firmu Rohlík a analyzovať jej stránky pomocou opísaných analytických nástrojov.

Štvrtá, a zároveň posledná kapitola, opisuje predložený návrh, vývoj a proces implementácie navrhnutého riešenia do firemného prostredia. Zároveň bude celý tento projekt analyzovaný pomocou metód opisovaných v druhej kapitole.

V závere bude uvedené krátke zhrnutie a taktiež vyhodnotenie celkovej práce a jej cieľov. Práca sa bude považovať za úspešnú, ak splní všetky ciele zadane v prvej kapitole.

Kapitola 1

Vymedzenie problému a ciele práce

Prvá kapitola v krátkosti uvádza problém, ktorým sa práca bude zaoberať, prečo sa ním bude zaoberať a čím prispeje jeho riešenie danej spoločnosti, v ktorej bude riešenie aplikované, a zároveň stanovuje ciele tohto projektu.

1.1 Vymedzenie problému

Jadro online obchodu Rohlík je ich sklad, z ktorého sa vydáva všetok potrebný tovar na dodanie svojim zákazníkom, pričom s veľkosťou takého skladu samozrejme prichádzajú rôzne problémy a poruchy pri bežnom zaobchádzaní.

Vzhľadom na to, že vedenie hľadalo riešenie na zefektívnenie komunikácie medzi zamestnancami v sklade, bola vymyslená mobilná aplikácia s názvom Rohcket, ktorej účelom je využiť práve jednoduchý prístup danej platformy na rýchlejšiu komunikáciu. Konkrétnejšie, ide o aplikáciu, v ktorej užívateľ vytvára tzv. tikety, respektíve záznamy, ktoré by ideálne mali opisovať nejaký problém (či už poruchu alebo nedostatok) v rámci skladu, ktorý je potrebné eliminovať.

1.2 Ciele práce

Cieľom a východiskom práce je analýza firmy, návrh a vytvorenie aplikácie na základe záverov analýz a samotná analýza návrhu a implementácie z hľadiska projektového manažmentu. Práca sa bude považovať za úspešnú, ak bude výsledný mobilný softvér aktívne používaný vo firemnom prostredí a zároveň bude analýza tohto projektu pokrytá v dostatočnej miere.

Kapitola 2

Teoretické východiská práce

Táto kapitola slúži na predstavenie rozličných základných termínov a nástrojov, ktoré budú v práci použité. Jej hlavnou myšlienkou je predstaviť čitateľovi teóriu v skratke a zoznámiť ho s primárnymi pojmami a nástrojmi, ktoré budú použité v rámci vypracovania predložených cieľov.

2.1 Základné termíny

Práca v tejto sekcii poukazuje na hlavné pojmy vyskytujúce sa v jej ďalších častiach. Jedná sa najmä o technické termíny vzhľadom na vývoj a implementáciu softvérovej aplikácie.

2.1.1 Dáta

Všeobecne sú v laickom zmysle dáta synonymom k slovu informácie. To však v podnikovej sfére nie je celkom pravda, keďže dáta bez kontextu nemusia niesť žiadnu informáciu.

V posledných rokoch patria dáta k najdôležitejším aktívam firiem, a blízka budúcnosť nenasvedčuje, žeby sa toto ponímanie malo zmeniť. Či už ide o dáta týkajúce sa mikro a makro okolia firiem, dáta o ich konkurencii alebo interné dáta, každá z týchto skupín im umožňuje robiť rôzne analýzy, ktoré im vedia pomôcť v budúcnosti, či už krátkodobej alebo dlhodobej. [19]

V dnešnej dobe je veľmi rozšírené zbieranie dát ohľadom správania svojich zákazníkov. K tým majú prístup najmä sociálne siete, keďže ľudia na nich zvyknú zdieľať svoj osobný život, svoje názory, myšlienky, fotky, atď. Tieto dáta sa buď následne predávajú iným firmám, ktoré to môžu využiť na generovanie zisku a implementáciu svojich systémov a výskumov, alebo ich samotná sociálna sieť môže využívať na generovanie personalizovaných reklám na základe preferencií zákazníka.

2.1.2 Programovací jazyk a kód

Ide o jazyk, ktorý môže užívateľ využiť na vytvorenie programu, resp. komunikovať s daným zariadením (či už s počítačom alebo mobilom) na to, aby dosiahol požadovanú funkcionality. Výstupom programovacieho jazyka je tzv. *kód*. Ten môže byť veľmi jednoduchý, napr. ak ide o jednoduchý výpočet dvoch vstupných prvkov x a y , ktoré sú následne sčítané, ale taktiež môže "poslať" človeka do vesmíru.

Skutočnosť je, že tak ako môže byť jedna myšlienka opísaná dvoma rôznymi vetami v rôznych jazykoch, tak isto môže byť program s rovnakou funkcionality napísaný dvoma rôznymi jazykmi a taktiež rôznym spôsobom. [21]

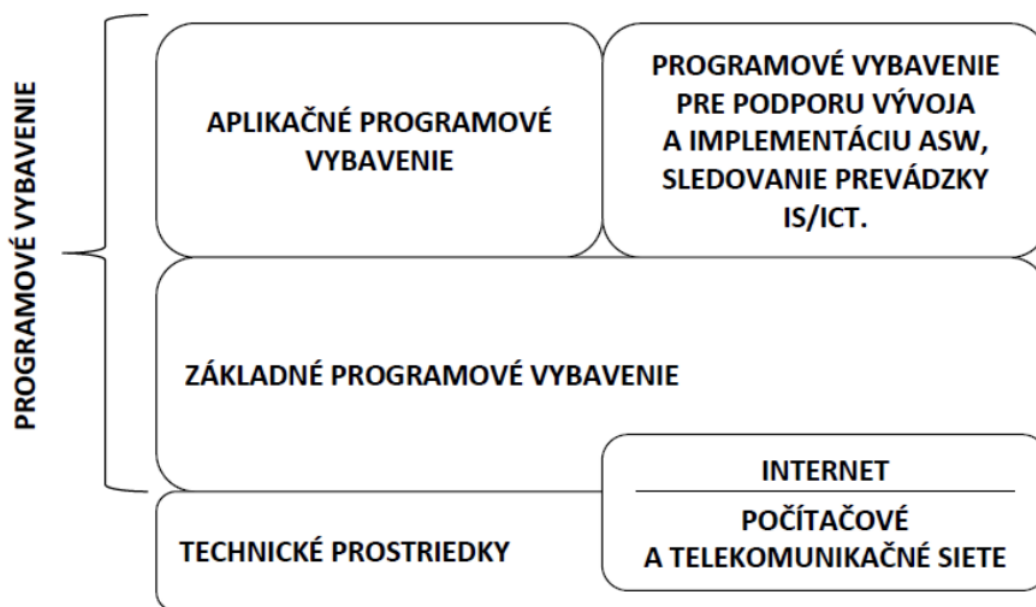
Existujú rôzne jazyky, ktoré majú iné využitie a sú preferované na rôzne funkcionality. Podľa dlhodobu udržiavaného indexu TIOBE, ktorý skúma popularitu jazykov pomocou rôznych zdrojov, najmä pomocou počtu výsledkov na rôznych vyhľadávateľoch. Medzi naj-

populárnejšie a najznámejšie momentálne patria jazyky v „v rodine C“ (C, C#, C++), Java, Python, Visual Basic a JavaScript, ktorý bude v tejto práci použitý na vývoj softvéru. [13]

2.1.3 Programové vybavenie (softvér)

softvér sa dá v podnikovej sfére rozdeliť do troch rôznych skupín:

- **Základné programové vybavenie (ZSW)** - zahŕňa softvér, ktorý ma na starosti komunikáciu programov, umožňuje prevádzku programov z ďalších skupín programov a ponúka prostriedky, ktoré umožňujú integráciu programov do väčších celkov.
- **Aplikačné programové vybavenie (ASW)** - slúži na spracovanie informácií a podporu podnikových procesov, pričom podkategórie zahŕňajú: *ASW transakčného charakteru* - softvér slúžiaci na obchodné transakcie, *ASW na podporu rozhodovania na všetkých úrovniach riadenia*, *ASW na rozvoj a inováciu produktu*, a nakoniec *infraštruktúrne aplikačné programové vybavenie*, ktoré slúžia na podporu rôznych aktivít v rámci firmy, napr. správa dokumentov a iného obsahu. Aplikácia vypracovaná v tejto práci zapadá do práve poslednej podkategórie, keďže bude podporovať chod skladov v rámci firmy.



Obr. 2.1: Základné skupiny prostriedkov tvoriace ICT [19]

2.1.4 Softvérová aplikácia

V dnešnom svete si človek môže pod spojením softvérová aplikácia predstaviť mnoho rôznych softvérov, najmä mobilné aplikácie sociálnych sietí ako Instagram, Facebook, atď.

Softvérová aplikácia je ale vo všeobecnosti všetok softvér, resp. program, ktorý sa dá niekde spustiť. Tento softvér je písaný v programovacích jazykoch špecializovaných na rôzne platformy (v prípade tejto práce ide o populárne platformy Android a iOS). Môže ísť o niečo tak jednoduché, ako je budík, stopky alebo kalkulačka. Naopak môže ísť aj o sofistikovaný softvér, medzi ktoré môžu patriť rôzne hry, ktoré vyžadujú veľký výkon, či už procesný alebo grafický. V oboch prípadoch však ide o softvérové aplikácie.

Najpopulárnejšími aplikáciami sa však dnes stávajú vyššie spomenuté sociálne siete a taktiež hudobné a video prehrávače, či už Spotify, Netflix alebo YouTube. V prípade tretej menovanej platformy, ide o aplikáciu, ktorá bola najskôr dostupná len prostredníctvom webových prehliadačov. Postupne sa však tento typ aplikácií preniesol do mobilných zariadení. Veľkou výhodou menovaných aplikácií je totiž vďaka jednotnej databáze a back-end technológiám ich všestrannosť, kde užívateľ môže väčšinou využívať aplikáciu na dvoch rôznych platformách bez toho, aby bol ukrátený o jej obsah. V tomto prípade, ten istý film, ktorý môže užívateľ pozerať prostredníctvom služby Netflix na svojom Android zariadení, môže taktiež pozerať na svojom prehliadači na osobnom počítači.

To však vyžaduje vývin aplikácie na dvoch rôznych platformách. Nie každá z nich tak musí byť naprogramovaná, keďže je na vývojároch, resp. objednávateľoch softvéru, či chcú, okrem mobilnej aplikácie, mať napríklad k dispozícii aj webovú stránku. Pre firmu Rohlík.cz bude momentálne vypracovaná len mobilná aplikácia, avšak v prípade záujmu by využitie frameworku React Native malo firme výrazne uľahčiť presun na webovú platformu.

2.1.5 Back-end

Veľmi zjednodušene ide o všetko, čo užívateľ pri využívaní stránky alebo softvéru nevidí. V dnešnom svete sú bežne používané webové stránky a aplikácie, ktoré majú dynamické dáta, resp. periodicky komunikujú s nejakým serverom, ktorý im tieto dáta posiela.

Dobrým príkladom sú „čatovacie“ aplikácie, ako Messenger alebo WhatsApp. Tieto programy vyžadujú neustálu aktualizáciu od serverov v prípade novej správy v schránke užívateľa. Ak je jeho zariadenie pripojené v nejakom momente na internetovú sieť, zmena bude okamžite odzrkadlená pri následnom použití tejto aplikácie, za čo môže práve neustála výmena dát medzi back-endom a front-endom (medzi serverom a rozhraním aplikácie). [15]

Prvotne musela však táto informácia, teda táto nová správa, byť najprv odoslaná niekým iným na server, kde bola následne uložená a poslaná príjemcovi do jeho aplikačného rozhrania.

Čo tieto informácie udržiava a manažuje, je teda back-end, v podstate softvér alebo aplikácia, ktorá je spustená na serveri, kde sa správy ukladajú. Front-end (strana, ktorú užívateľ vidí) s touto aplikáciou komunikuje pomocou rôznych príkazov, ako v prípade odosielania a prijímania nového obsahu.

V tejto práci bude v rámci serveru využitá aj databáza, ktorá bude opísaná v nasledujúcej podkapitole. Práve back-end je využívaný na jej kontrolu a ukladanie/sťahovanie dát.

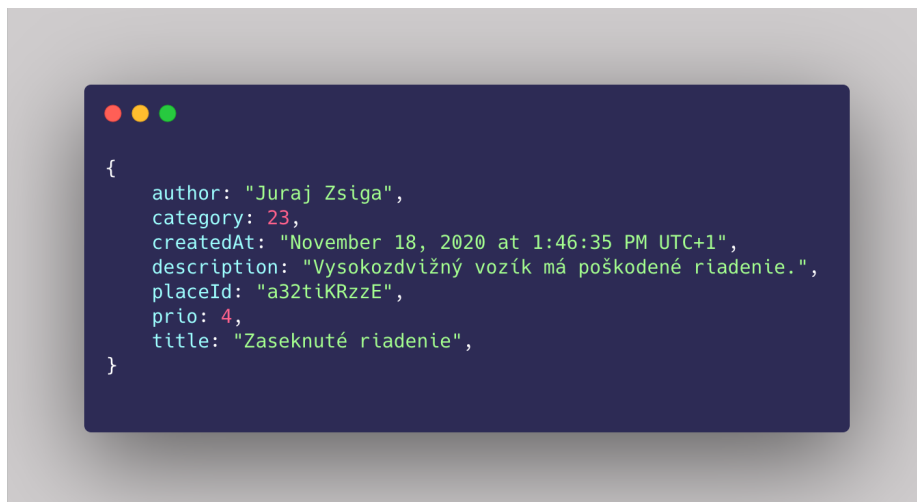
Z hľadiska programovania je možné využiť niekoľko populárnych jazykov, či už Java, Python, PHP. V prípade tejto práce však pôjde o JavaScript, ktorý v podstate pokryje celý „stack“, takže front-end a back-end.

2.1.6 Databáza

Databáza uschováva *dáta* v organizovanej podobe. Z tohto hľadiska je veľmi dôležitá jej architektúra, ktorá by mala byť prehľadná a jednoducho ovládateľná, na čo sa v detaile zameriava *dátový manažment*.

Je dôležité si uvedomiť, že databáza zväčša neuchováva jednoznačne čitateľné informácie, ale surové fakty, na základe ktorých vie vývojár vyhodnotiť nejaký stav. To znamená, že veľa dát nemusí vo svojej holej podobe dávať bez kontextu žiaden zmysel, preto je veľmi dôležitá dokumentácia, ktorá by správne mala opisovať význam daných štruktúr. [16]

V rámci tejto práce bude databáza slúžiť primárne na ukladanie jednotlivých záznamov, ktoré užívateľ aplikácie vyplní na základe príslušného problému, ktorý sa v sklade vyskytne. Každý záznam bude v tomto prípade uložený pod unikátnym identifikátorom (*id*), ktorý je zložený z náhodne generovaných znakov a jeho obsah bude pozostávať z niekoľkých polí (ang. *field*), ako napr. *description*, *title*, *createdAt*, atď. (bližšie informácie ohľadom štruktúry budú v samotnej implementácii).



Obr. 2.2: Príklad štruktúry záznamu (zdroj: vlastná tvorba)

Na obrázku 2.2 je práve možné vidieť niekoľko polí, ktoré nie je možné vyhodnotiť bez nijakého kontextu. Napríklad pole s názvom *category*, ktoré má v sebe uložené číslo 23. V tomto prípade spadajú tieto dáta pod kategóriu s číslom 23, pričom však informácia o tom, ktorá kategória nesie číslo 23, je uložená inde.

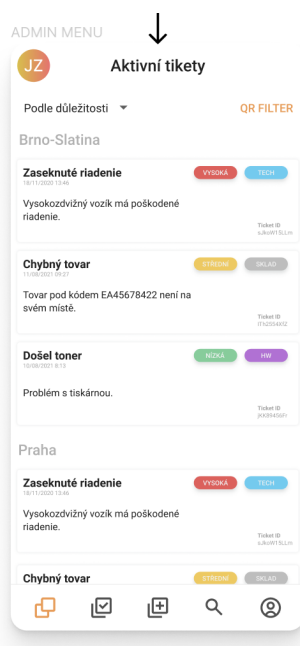
V tejto práci sa bude využívať tzv. NoSQL databáza, resp. databáza, s ktorou sa nebude komunikovať pomocou iného jazyka ako je SQL. Výhodou tohto typu databázy je, že nie je relačná, takže vo svojej podstate nie je nutné dodržiavať rovnaké, preddefinované štruktúry dát. To robí túto štruktúru flexibilnú, keďže je na používateľoch tejto databázy, či budú stále používať svoje preddefinované modely. V prípade tejto práce pôjde o využitie MongoDB v kombinácii s back-end technológiou Express.

2.1.7 Front-end

V predošlej sekcii bol opísaný back-end, to je časť aplikácie, ktorú užívateľ nevidí. Opakom tejto časti je tzv. front-end, čo je v podstate užívateľské rozhranie.

Existujú aplikácie, ktoré sa skladajú len z front-endu, takže ich dáta a rozhranie sú staticky naprogramované a neprijímajú žiadne dynamické dáta. Veľká časť kódu, ktorý bude vypracovaný pre túto prácu na vytvorenie aplikácie Rohcket, bude práve pozostávať najmä z front-endovej časti.

Väčšina ľudí sa vo svojich životoch stretne len s touto časťou aplikácie. Ide o niečo, čo sa deje na užívateľovom zariadení, či už je to displej alebo iný hardvér. Ako už bolo spomenuté, navrhovaný softvér bude zložený najmä z kódu pre toto užívateľské rozhranie. To totiž bude vyhodnocovať dáta, ktoré príjme z databázy pomocou využitia *back-endu*, a následne ich dynamicky zobrazí, poprípade bude využitý na zobrazenie niečoho iného, ako v obrázku 2.3. Zobrazovanie týchto tiketov, ktoré budú prebraté z databázy, bude hlavnou úlohou tohto programu určeného pre sklad.



Obr. 2.3: Dynamicky zobrazené tikety v aplikácii (zdroj: vlastná tvorba)

2.1.8 Framework

Framework je v podstate zbierka rôznych pred-programovaných metód a knižníc pre daný programovací jazyk, ktoré by programátorovi mali teoreticky uľahčiť prácu pri písaní v tomto jazyku. Tým pádom sa považujú za efektívnejšie aj v rámci biznisu, pretože s ich pomocou môže tím vývojárov naprogramovať aplikáciu rýchlejšie, čím ušetrí zdroje, či už finančné alebo časové. [31]

Rýchlejšie programovanie pomocou týchto nástrojov spočíva najmä v možnosti používania toho istého kódu na viacerých miestach, tzv. *code reusability*, čo v preklade znamená „miera znovu použiteľnosti kódu“. Okrem iného, podstata frameworku spočíva aj v lepšej kontrole toku dát cez samotnú aplikáciu. Najpopulárnejšími v rámci JavaScriptu sú Angular, Vue.js, React a ich natívne verzie, pomocou ktorých sa dajú vytvoriť aplikácie nielen na webovej platforme, ale na mobilných platformách. [26]

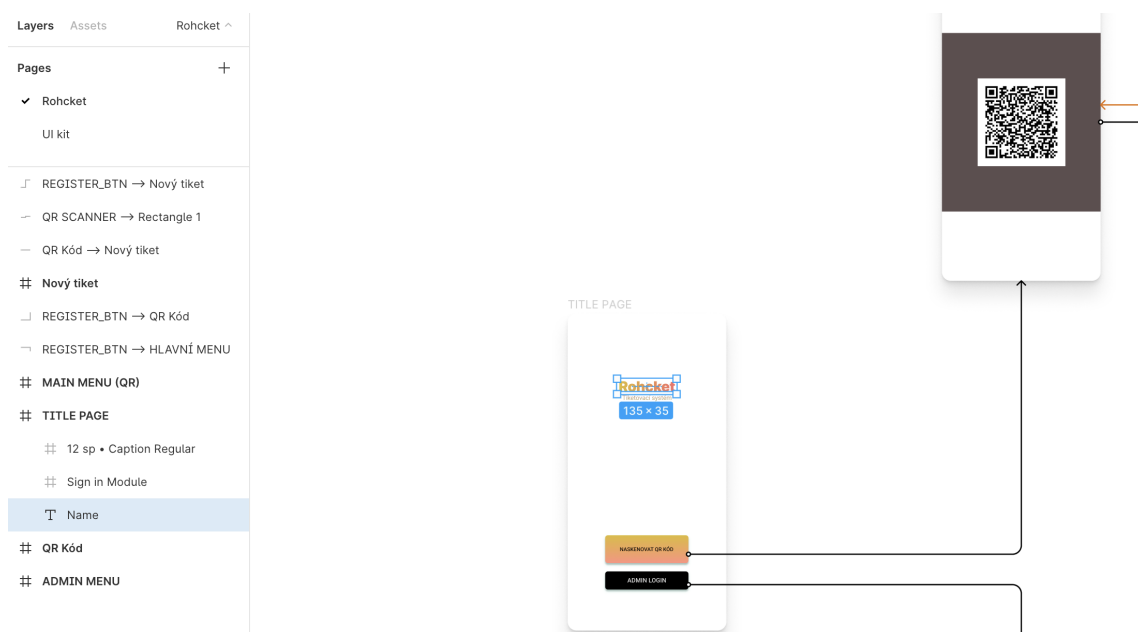
Práve v tejto práci pôjde o využitie frameworku React Native, pomocou ktorého je možné naprogramovať aplikáciu pre najrozšírenejšie mobilné platformy, Android a iOS.

2.2 Pomocné nástroje

2.2.1 Figma

Figma je nástroj určený na vizuálny design aplikácií (či už webových alebo mobilných). Človek môže buď začať s prázdny m plátnom, alebo má na výber zo šablón, ktoré boli vytvorené inými užívateľmi Figma. V rámci práce bola táto aplikácia použitá pred písaním kódu.

V minulosti sa nebral ohľad na design tak ako v dnešnej dobe, kedy je dobre premyslená aplikácia s myšlienkou ponechať užívateľovi čo najlepší zážitok viacmenej štandardom pre vývojárov, preto je veľmi dôležité začať s vhodným designom, ktorý by mal byť ucelený, a v ktorom by sa mal samotný užívateľ aplikácie dobre orientovať bez toho, aby potreboval nejakú značnú pomoc. [18]



Obr. 2.4: Rozhranie aplikácie Figma

2.2.2 Expo

Ide o platformu pre vývojárov zameranú na programovanie vo frameworkoch React a React Native, pomocou ktorej je veľmi jednoduché vyvíjať aplikáciu pre rôzne platformy. Výhodou Expo je, že nevyžaduje veľkú mieru znalosti, ak ide o inicializáciu pomerne jednoduchšieho softvérového projektu, a zároveň je možné spustiť rovnaký kód na viacerých zariadeniach, či už ide o zariadenia používajúce operačný systém Android, iOS, alebo dokonca novšie verzie OS Windows. [6]

Hlavným dôvodom výberu tejto platformy na vypracovanie aplikácie Rohcket je ten, že pohodlne spĺňa podmienky návrhu a implementácie aplikácie, ktorá bola vyvinutá v rámci tejto práce.

2.2.3 GitHub

GitHub je populárny portál používaný vývojármi po celom svete. Slúži na ukladanie kódu a jednoduchú spoluprácu medzi viacerými užívateľmi, či už ide o súkromné, „hobby“ projekty alebo o komerčné aplikácie. Portál funguje na základe tzv. repozitárov, v ktorých sú uložené jednotlivé projekty. Každý repozitár by sa mal skladať z dokumentácie a uloženého kódu, ktorý môže byť veľmi jednoducho udržiavaný a menený prostredníctvom systému *git*.

GitHub má v sebe zakomponovaný agilný tabuľkový systém, v ktorom je možné plánovať sprinty, s názvom GitHub Issues. Ide o jednoduchú verziu populárnych agile systémov, ako je napríklad Trello alebo Jira.

Dôvod výberu tohto nástroja na vývoj je veľmi jednoduchá orientácia a zároveň samotná integrácia GitHub Issues, pomocou ktorej je projekt rozvrhnutý do menších častí, čo dodáva užívateľovi lepší prehľad nad svojim projektom.

2.3 Back-end technológie

Táto sekcia má na starosti predstaviť nástroje použité pri vývoji aplikácie z hľadiska back-end technológií. V tejto časti budú spomenuté tri časti tzv. „MERNG stack“, čo je vývojový systém zložený zo štyroch rôznych elementov. Z back-end technológií sú to práve databáza MongoDB (M) a back-endových technológií Node.js, Express a GraphQL.

2.3.1 JavaScript

Ide o populárny programovací jazyk, ktorý vznikol v roku 1995 v úmysle zaviesť aplikácie do webových stránok, konkrétne pre internetový prehliadač Netscape Navigator. Tento jazyk však okrem názvu nemá nič spoločné s iným známym jazykom Java. [21]

Zo začiatku bolo tento jazyk, resp. kód tohto jazyka, možné spustiť len v internetovom prehliadači užívateľa, preto ani nebol veľmi využívaný na vývoj iného softvéru ako webových stránok. Po nejakom čase bol však vyvinutý Node.js, ktorý dovoľuje vývojárom programovať aplikácie na iné platformy ako len prehliadače. Týmto spôsobom funguje niekoľko populárnych počítačových programov, ako sú Discord, Spotify a iné. Boli naprogramované JavaScriptom, avšak nie je potrebné spúšťať ich prostredníctvom Google Chrome alebo Mozilla Firefox, ide totiž o samostatnú aplikáciu, ktorá je schopná byť sebestačne spustená. Mimo iného to dovolilo programátorom používať JS ako back-endové technológie. Táto taktika, alebo „stack“, sú výhodné hlavne tým, že vývojár nie je nútený sa naučiť dva rôzne jazyky, keďže mu stačí vedomosť JavaScriptu.

Či už ide o front-end alebo back-end, k tomuto jazyku sa po rokoch vymysleli rôzne programovacie knižnice a šablóny (tzv. framework), prostredníctvom ktorých sa zjednodušilo programovanie webových stránok, keďže užívateľovi prinášajú viacero predprogramovaných nástrojov na vývoj. Najpopulárnejšími front-endovými sú React, Ember a Vue.js. [7]

So stúpajúcou popularitou boli tieto frameworky rozvinuté do takej miery, že na nich boli postavené rozhrania na vývoj mobilných aplikácií. Jedno z týchto rozhraní je práve založené na spomenutej knižnici React a nesie názov React Native.

Na back-end sa využívajú najmä Express, Gatsby či Next.js. Prvá menovaná technológia je použitá v tejto práci, a bude predstavená v tejto časti.

Treba podotknúť, že JavaScript nie je len jazyk na back-end, ale aj na front-end. Jeho využitie v rámci nadstavby React Native bude predstavené v ďalšej podkapitole zameranej na front-end.

2.3.2 Node.js

Ako už bolo spomenuté, zo začiatku bolo možné spustiť kód vytvorený v programovacom jazyku JavaScript len prostredníctvom internetového prehliadača, čo znamenalo, že nebol použiteľný ako serverový, resp. jazyk použiteľný na back-end. Táto medzera otvorila priestor pre vytvorenie technológie, ktorá by dovoľila vývojárom programovať aplikácie aj na pozadí, resp. na komunikáciu so serverom.

Na tento účel bola vytvorená aplikácia Node.js. Ide vlastne o technológiu V8, tzv. „JavaScript engine“ (kompilátor jazyka JavaScript), vytvorený firmou Google, ktorý je používaný v ich internetovom prehliadači Google Chrome. Tento „motor“ bol v podstate vybratý z ich prehliadača a nasadený na server, čo otvorilo vývojárom nové možnosti z hľadiska vývoja tzv. server-side aplikácií, keďže tento vynález im dovoľuje využitie ďalšieho jazyka.



Obr. 2.5: Využitie Node.js v praxi [17]

Na obrázku 2.6 je možné vidieť niekoľko využití danej technológie. Napriek tomu, že bol zo začiatku svet vývojárov ohľadom nej skeptický, si Node.js svoju popularitu vytvoril, keďže

sa svojim širokým pokrytím funkcionalít sa preukázal ako silný konkurent iných podobných serverových softvérov.

Hlavnou výhodou je to, že ak už programátor ovláda JavaScript na úrovni front-end, môže využiť svoje znalosti na programovanie serverových aplikácií bez toho, aby sa musel učiť taký objem nových poznatkov, ako napríklad ďalší jazyk, či už Python, PHP a iné.

2.3.3 Express

Je to nadstavba vyššie definovanej technológie Node.js. Jej účelom je zjednodušiť celkové programovanie prostredníctvom Node.js a tým pádom aj zefektívnenie vývoja serverových aplikácií pre daný projekt. Prostredníctvom Express je možné vytvárať cesty, ktorými má aplikácia ísť, keď sa k nej dostanú požiadavky HTTP. Práve tieto cesty sú preddefinované na základe zloženia týchto požiadaviek. [34]

2.3.4 GraphQL

GraphQL je jazyk, pomocou ktorého je možné sa dotazovať na back-end aplikácie. Jeho výhodou je lepšia špecifikácia dát, ktoré potrebuje front-end dostať. Príklad takého využitia môže byť rozdiel zobrazenia produktu v e-shope na mobilnom zariadení a na internetovej stránke.

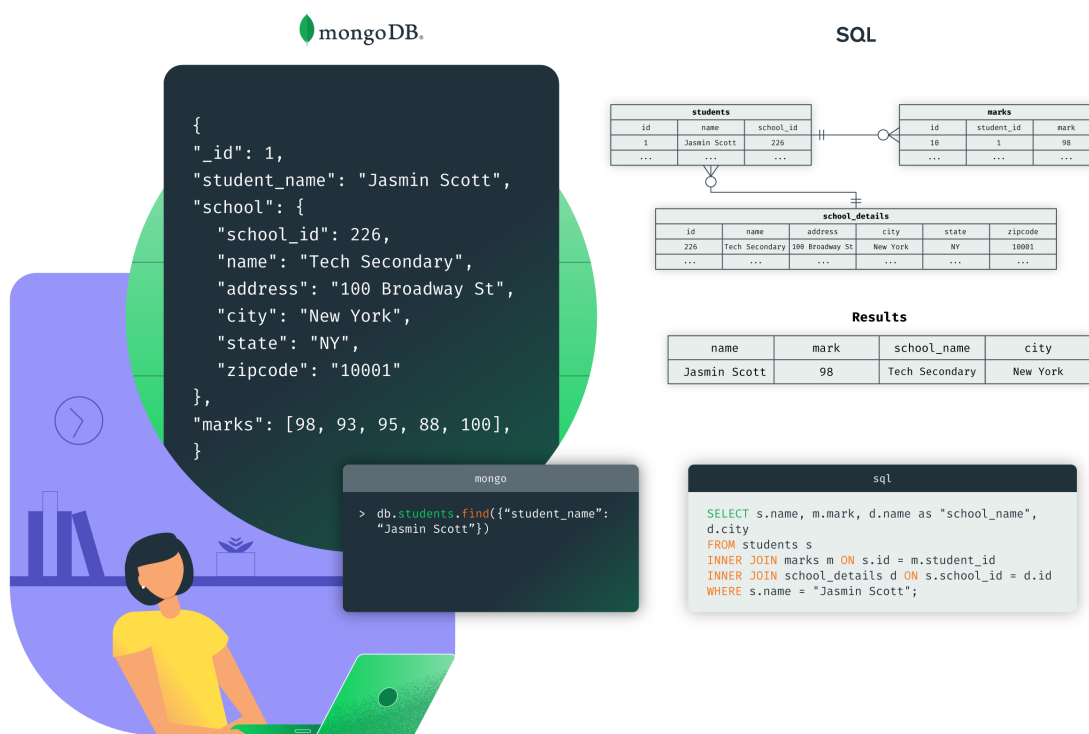
V prípade mobilu by mohlo ísť o jednoduché dáta, ako sú názov produktu, odkaz na obrázok a cena produktu. V prípade stránky by mohlo ísť o sofistikovanejšie dáta, ktoré by okrem tých, ktoré zahŕňa aj mobilná integrácia, mohli byť napríklad kód produktu, stav zásob, atď. Na tomto príklade je vidno, že na zobrazenie produktu je potrebných menej informácií, ako na zobrazenie toho istého produktu prostredníctvom stránky. GraphQL dokáže z rovnakého objektu daného produktu v databáze teda stiahnuť len dáta relevantné k danej platforme bez toho, aby sťahoval prebytočné dáta, čím celkovo zrýchľuje celú aplikáciu.

2.3.5 MongoDB

Ide o dokumentovú databázu, čo znamená, že každý záznam je samostatný dokument alebo vo svojej podstate objekt. Tieto objekty majú podobné zloženie ako objekty v tzv. jazyku JSON, teda *JavaScript Object Notation*, takže sú v podstate písané v jazyku JavaScript.

Výhodou tejto databázy je, že na vytvorenie vnorených polí v dokumente nie je nutné vytvárať ďalšie tabuľky ako pri relačných databázach. [34]

Dokumenty sú ukladané do kolekcií, ktoré sú väčšinou pomenované podľa toho, čo je do týchto kolekcií ukladané (takže obrázok bude ukladaný do kolekcie *obrázky*, video do kolekcie *videá*, atď.). Avšak, je samozrejme na užívateľovi, ako tieto kolekcie pomenuje. Čo sa týka pomenovania dokumentov, ak to užívateľ nedefinuje, je pre každý dokument vygenerovaný náhodný identifikátor, *index*, skladajúci sa zo znakov abecedy a čísel. To umožňuje vývojárom ukladať veľké množstvo rovnakého typu dokumentov do kolekcií. Pri menšom počte dokumentov môže byť táto vymoženosť zanedbateľná, avšak pri obrovskom objeme užívateľov, ktorý ma napríklad Facebook, je toto automatické generovanie priam nevyhnutné.



Obr. 2.6: Rozdiel medzi databázami MongoDB a SQL [1]

2.4 Front-end technológie

2.4.1 React

React bol vytvorený firmou Facebook za účelom zjednodušenia vývoja webových stránok a aplikácií. Vo svojej podstate ide o knižnicu, ale s pridaním ďalších rozšírení a ďalších knižníc, ktoré boli pre React naprogramované, spolu tieto nástroje tvoria jednu z najpopulárnejších nadstavieb, teda frameworkov jazyku JavaScript.

Zo začiatku bol React používaný na vytvorenie internetových stránok, ale postupom času bol vyvinutý do iných foriem, ako Electron (desktopové aplikácie) alebo React Native (mobilné aplikácie).

Základ každej aplikácie vytvorenej týmito nástrojmi sú tzv. „komponenty“. Jednotlivé stránky sú práve zložené z viacerých komponentov, ktoré môžu byť programované buď staticky alebo dynamicky. Práve dynamické programovanie komponentov ponúka vývojárom veľké možnosti prepojenia ich stránok so serverom (teda prepojenie front-end <-> back-end). Do týchto stavebných častí sú dáta zo serveru posielané ako tzv. „props“ (skratka pre ang. „properties“, teda rekvizity). Tieto *props* sú väčšinou práve dynamicky zobrazované informácie na obrazovkách užívateľa.

Obrázok 2.3 presne ukazuje rovnaké stavebné jednotky, *komponenty* (teda tikety), ktoré nesú rovnaké *props* (napr. *titul tiketu*), avšak každý z týchto titulov sa líši. Každopádne sa jedná o rovnakú stavebnú jednotku skladajúcu sa vždy z rovnakého dizajnu, akurát z odlišného obsahu.

2.4.2 React Native

Ako už bolo spomenuté, ide o framework písaný v jazyku JavaScript, založený na základe webovej nadstavby React. React Native je konkrétne využívaný na písanie natívnych aplikácií pre rôzne platformy, v prípade práce pre platformy Android a iOS. Bol vyvinutý firmou Facebook a vydaný v roku 2015, pričom dôvodom jeho existencie je hlavne zrýchlenie celkovej aplikácie a teda aj celkového UX (*user experience*). [27]

Dôvod výberu práve tohto rozhrania je predošlá praktická skúsenosť a taktiež výhoda v tom, že je možné použiť jeden zdrojový kód na obe mobilné platformy naraz, čo v podstate efektívne kráti pracovný čas o polovicu, keďže nie je potrebné vymýšľať dva rôzne kódy pre jednotlivé operačné systémy.

2.5 Analytické nástroje

Na úspešnú analýzu stavu vybranej spoločnosti je nutné vybrať niekoľko overených metód a techník. V tejto časti kapitoly budú stručne predstavené modely, ktoré budú v analytickej časti práce využité práve na rozbor spoločnosti Rohlík.

2.5.1 SLEPT

Odhaľuje budúci vývoj vonkajšieho prostredia firmy, kvôli čomu sa taktiež niekedy nazýva *analýza širšieho vonkajšieho prostredia*. [20]

Metóda sa skladá z piatich faktorov, ktoré medzi sebou môžu občas aj úzko súvisieť [4]:

- *Social*. Ide o spoločenské faktory, väčšinou cieľových skupín, ako sú demografický vývoj populácie, úroveň vzdelávania, sociálne postavenia, percentuálne zastúpenie pohlaví, atď.
- *Legal*. Právne faktory, ako sú stabilita právneho systému, v ktorom sa podnik nachádza, právne normy, priehľadnosť a miera ich zložitosti, atď.
- *Economic*. Mikroekonomické a makroekonomické faktory, ako sú vplyv miestnej politiky na hospodárstvo, zároveň tu spadá aj miera regulácie hospodárstva, a taktiež monetárna politika daného štátu. Medzi iné sa tu dá zaradiť aj daňové zaťaženie a cena práce v okolí, s čím môže súvisieť aj spoločenský faktor, ako je napríklad sociálne postavenie.
- *Political*. Politické faktory zahŕňajú stav a stabilitu politického systému, vojny, obmedzenia, korupciu, atď.
- *Technological*. Ide o vývoj v technológiách, vynálezy a objavy, ktoré by mohli mať vplyv na podnik.

Použitie tejto metódy spočíva v tom, že sa analyzuje minulý stav a súčasný stav firmy, a na základe toho sa predpovedá budúci stav, ktorý vyhodnotí, či sa firme vyskytne buď príležitosť, poprípade nejaká hrozba.

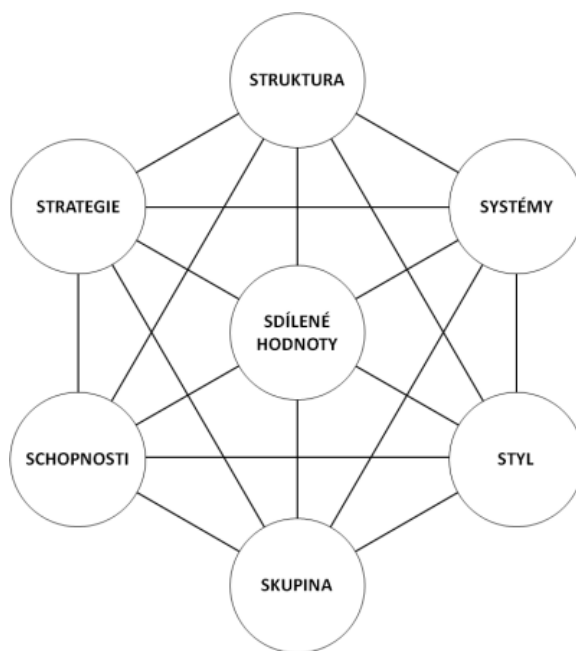
2.5.2 Porterova analýza

Jedná sa o analýzu konkurenčného prostredia vyvinutú Michaelom Porterom. Ten, vo svojej knihe *Konkurenční strategie* [28], opisuje model piatich síl/hrozieb:

1. *Hrozba intenzívnej konkurencie*. Ide o počet konkurentov v odvetví a zároveň o ich schopnosť konkurovať analyzovanému podniku. Čím viac je konkurentov alebo čím väčšia je konkurenčná sila iných podnikov, tým je horšie sa pre daný podnik v odvetví presadiť. Ak je však táto sila nízka, podnik môže svoju silu v odvetví využiť rôznymi spôsobmi (napr. zvyšovaním cien).
2. *Hrozba nových firiem v odvetví*. Čím ľahšie, resp. čím menšie množstvo zdrojov potrebné na to, aby sa podnik vedel dostať do odvetvia, tým zraniteľnejší môže analyzovaný podnik byť. Preto je výhodné, ak spoločnosť podniká v oblasti, ktorá má vysokú bariéru vstupu pre potenciálnych konkurentov.
3. *Hrozba rastúcej kompetencie dodávateľov*. Ide o niekoľko faktorov naraz. Aký produkt sa dodáva, je jeho množstvo malé alebo veľké, aký je po ňom dopyt, koľko by stálo vymeniť dodávateľa, atď. Čím menej je dodávateľov pre dané odvetvie, tým viac je spoločnosť odkázaná na dodávateľa, ktorého má. V tom prípade je sila v jeho rukách, pretože si môže určovať vlastné ceny. V opačnom prípade má práve silu analyzovaná firma, keďže môže minimalizovať náklady za účelom zisku.
4. *Hrozba rastúcej kompetencie zákazníkov*. Môže ísť o zvýšený tlak zákazníkov na rôzne aspekty výrobku – kvalita, cena, objem, čo môže vyvolať napätie v odvetví. Ak na trhu existuje nejaký substitút, tak malý, organizovaný počet zákazníkov môže pre podnik predstavovať hrozbu, keďže zákazník má v tomto prípade väčšiu vyjednávaciu silu. Pri väčšom počte zákazníkov a žiadnom substitučnom produkte platí opak.
5. *Hrozba substitučných produktov*. Iné produkty, ktoré dokážu nahradiť produkty analyzovanej firmy, môžu ohroziť jej predaj na základe rôznych faktorov, ako môže byť cena, kvalita, a v dnešnej dobe aj značka.

2.5.3 Rámec 7S

Vyvinutý zamestnancami firmy McKinsey, ide o metódu, ktorá skúma sedem faktorov úspechu. Na základe tejto analýzy by mal výstup predpokladať úspech (alebo neúspech) analyzovanej spoločnosti. [29]



Obr. 2.7: Rámec 7S od firmy McKinsley [2]

Vysvetlivky pre dané hodnoty [29]:

- *štruktúra* – ide o typ prerozdelenia úloh a kompetencií v rámci spoločnosti
- *systémy* – všetky informačné procedúry prebiehajúce v rámci firmy (formálne a neformálne)
- *štýl* – typ vedenia firmy (autoritatívny, demokratický alebo laissez - faire)
- *spolupracovníci* – ako sú zamestnanci motivovaní, či sú splnené ich základné potreby, či sa s nimi zachádza kompetentne
- *schopnosti* – všeobecné tempo rozvoja firmy a rýchlosť adaptácie vzhľadom na interné a externé zmeny

- *stratégia* – väčšinou je určená samotnou víziou, resp. dlhodobým smerovaním firmy k predurčenému cieľu, ktorý je možný zistiť buď z dôverných materiálov danej firmy alebo z celkového trendu/atmosféry v spoločnosti
- *zdieľané hodnoty (shared values)* – súhrn hodnôt a všeobecných predstáv ohľadom firmy samotnej, resp. nejaká kultúra firmy

2.5.4 SWOT analýza

Súčasť strategického plánovania projektu. Ide o odhalenie štyroch základných ukazovateľov analyzovaného subjektu:

- *STRENGTHS* – silné stránky
- *WEAKNESSES* – slabé stránky
- *OPPORTUNITIES* – príležitosti
- *THREATS* – hrozby

Niektorí ľudia sa k týmto hodnotám dostávajú prostredníctvom myšlienkových máp, ktorých účelom je odkryť čo najviac súvisiacich vecí s danou tematikou. Vďaka tejto veľmi základnej metóde je človek schopný určiť, na čo by sa mal projekt zamerať, resp. čoho by sa mal vyvarovať, taktiež pomocou silných a slabých stránok vie efektívnejšie prerozdeliť úlohy pomedzi členov tímu. [33]

2.6 Projektové metódy

Časť má v úmysle predstaviť čitateľovi súbor metód použitých pri analýze samotného návrhu a implementácie mobilnej aplikácie v prostredí firmy Rohlík. Pôjde o kratšie opisy, ktoré by mali načrtnúť základné informácie ohľadom použitých metód.

2.6.1 Lewinov model

Lewinov model sa dá všeobecne zhrnúť do troch základných krokov [30]:

1. Fáza rozmrazenia - príprava zmeny
2. Prechod na novú úroveň - uskutočnenie tejto zmeny
3. Fáza zmrazenia - finalizácia nových výsledkov

Tento model slúži na riadenie zmien, od ich navrhnutia až po monitorovanie. Lewin tvrdil, že pred tým, než implementácia zmeny započne, je dôležité sa spýtať niekoľko otázok [30]:

- **Ktoré** faktory iniciujú túto zmenu, **aká** veľká je ich sila?
- **Ako** vyzerá požadovaný konečný stav, ktorý chceme dosiahnuť?
- **Ktoré** sily budú za a **ktoré** budú proti zmene?
- **V ktorej** sfére firmy táto zmena nastane?
- **Ako** sa táto zmena uskutoční?
- **Ako** celý proces implementácie zmeny dopadol?

Detailnejšie je tieto tri kroky možné rozdeliť do etáp, ktoré zahŕňajú analýzu pred a po zavedení zmeny. Vzhľadom na prvú etapu ide o identifikáciu možných zlepšení v rámci firmy [30]:

- a) analytická etapa
- b) návrhová etapa - vytvorenie modelu zmeny, stanovenie agenta zmeny a určenie firemných procesov, ktoré budú zmenou ovplyvnené

- c) realizačná etapa
- d) spätné vyhodnotenie uskutočnenej zmeny

V poslednej etape je zmena monitorovaná, ak by nebola dostatočná, došlo by k optimalizácii zmeny, ak by dostatočná bola, dochádza k zmrazeniu a finalizácii implementácie tejto zmeny.

2.6.2 Metóda RIPRAN

Na analýzu rizík bude využitá metóda RIPRAN, ktorá bola vyvinutá pre malé až stredné firmy doktorom Branislavom Lackom.

Tak isto ako iné metódy analýzy rizík, aj táto slúži na ich identifikáciu a odhalenie. Avšak samotné ohodnotenie jednotlivých hrozieb nepozostáva z číselných hodnôt, ale z verbálnych. Na svojich stránkach Lacko predstavuje tri sústavy, podľa ktorých môžu byť riziká ohodnotené. [25]

Sústava 2x2x2 je používaná na približné ohodnotenie rizík, takže sa používa v prípade, kde chce projektový tím len zľahka ohodnotiť hrozby daného subjektu.

Ďalšia sústava, 3x3x3, je používaná pri bežných projektoch, poprípade pri väčších projektoch, v ktorých však projektový tím nedisponuje vhodným objemom štatistických podkladov.

Poslednou a najdetailnejšou sústavou je sústava 5x5x5, ktorá je využívaná práve na ťažšie a väčšie projekty, v ktorých sú informácie ohľadom potenciálnych hrozieb k dispozícii.

Metóda RIPRAN sa skladá z týchto fáz [25]:

1. Príprava analýzy rizika. V tomto kroku je vybratý tím na analýzu rizík, je vybraná sústava, ktorá bude použitá, a taktiež je vytvorený dokument, resp. šablóna, do ktorej sa budú hrozby vyhodnocovať.
2. Identifikácia rizika. Podľa názvu ide o fázu, v ktorej sú hrozby identifikované a odhalené.
3. Kvantifikácia rizika. Fáza zahŕňa ohodnotenie pravdepodobnosti hrozby. Stupnica ohodnotenia závisí od vybranej sústavy.
4. Odozva na riziko. Po ohodnotení pravdepodobnosti prichádza ohodnotenie dopadu rizika na projekt, a aký scenár by pri výskyte tohto rizika nastal.

5. Celkové zhodnotenie rizika. Po splnení predošlých dvoch fáz prichádza ohodnotenie hrozby, resp. či je nutné na riziko vymyslieť nejaké opatrenia.
6. Sledovanie a vyhodnocovanie rizík v projekte. Koniec celkovej analýzy rizík, prípadné monitorovanie rizík, ktoré sa môžu vyskytnúť počas exekúcie projektového plánu. Ak by tieto nové riziká nastali, proces sa opakuje. V prípade nekončiaceho sa projektu, resp. dlhodobého projektu, môže teda ísť o cyklickú analýzu a monitorovanie.



Obr. 2.8: Fázy metody RIPRAN [25]

2.6.3 Metóda SCRUM a sprinty

Ide o agilnú metodiku riadenia projektu. Na začiatku celého projektu sa zvolia hlavné činnosti, resp. funkcionality, ktoré má produkt obsahovať. Tieto činnosti sú rozdelené podľa priority na niekoľko tzv. sprintov.

Sprinty sú v podstate časové obdobia (zvyčajne dva týždne), ktoré majú vždy stanovený určitý cieľ, ktorý má projektový tím počas daného sprintu dosiahnuť. Účelom tejto metódy je rozdeliť celý projekt na menšie časti, ktoré rozdeľujú celý projekt do niekoľko dvojtýždňových blokov. [32]

V programovaní a vývoji softvérových aplikácií je táto metodika jedna z najpoužívanějších. Písanie kódu a aktualizácia aplikácie je veľmi premenlivá operácia, ktorá od zamestnancov vyžaduje veľkú mieru prispôsobivosti. Z hľadiska užívateľa môže veľakrát ísť o pridanie novej funkcionality zo dňa na deň, na čo by bola napríklad metóda PERT zbytočne dlhá. V prípade, žeby šlo o menšiu funkcionality, ktorú je možné pridať do dvoch týždňov, je práve rozdelenie tejto úlohy na viacero činností v rámci jedného sprintu výhodné riešenie z pohľadu projektového plánovania. [32]

Vzhľadom na to, že aj v tejto práci ide o softvérovú aplikáciu, bude návrh časovo rozložený pomocou scrum metódy, pričom budú dané ciele rozdelené do niekoľkých dvojtýždňových sprintov, ku ktorým sú priradené jednotlivé ciele.

Kapitola 3

Analýza problému a súčasnej situácie

Kapitola 3 má za úlohu predstaviť firmu Rohlík, v ktorej bude aplikácia Rohcket implementovaná. Pôjde o jej základné informácie, postavenie na trhu, po čom bude následne spoločnosť analyzovaná prostredníctvom nástrojov opísaných v predošlej kapitole. Výsledok tohto rozboru by mal preukázať dôvod návrhu a implementácie nového softvéru do každodenného chodu v rámci skladov tohto podniku, ktoré sú pre typ takých spoločností dôležité, keďže kvalita skladovania a zaobchádzanie so skladovaným tovarom je dôležitou súčasťou životného cyklu tejto služby.

3.1 Základné informácie

Obchodná firma:	VELKÁ PECKA s.r.o.
Dátum vzniku:	21.5.2014
Právna forma:	Spoločnosť s ručením obmedzeným
Sídlo:	Sokolovská 100/94, Karlín, 186 00 Praha 8
IČO:	03024130



Obr. 3.1: Logo Rohlík.cz [9]

3.2 Predstavenie firmy

Firma Velká Pecka s.r.o., známejšia pod názvom Rohlík.cz, vznikla v roku 2014. Najväčší podiel firmy drží majiteľ Slevomatu a DámeJídlo.cz, Tomáš Čupr. V rozhovore pre Seznam Zprávy z marca roku 2020 mimo iného vrel, že založeniu online supermarketu predchádzali myšlienky, ako:

„Říkal jsem si, že když doručujeme teplou pizzu do hodiny v DámeJídlo.cz, co může být těžkého na tom, doručit studené mléko?“. [24]

Služby sú v súčasnosti poskytované vo viacerých veľkých mestách ČR, vrátane Prahy a Brna. Okrem Českej Republiky sa firma rozšírila aj do zahraničia pod inými názvami, napr. v Maďarsku pod názvom Kifli.hu a v Rakúsku pod názvom Gurkerl.at.

V Česku je podľa poslednej výročnej správy zamestnaných zhruba 300 ľudí. Za posledný rok boli vybudované nové sklady, ktoré zároveň zvýšili počet zamestnancov v rámci ČR.

Presný počet nie je známy, ale jednatel tejto spoločnosti očakáva nový prírastok, ktorý by mal nakoniec znamenať zhruba dvojnásobný počet zamestnancov. [24]

Okrem iného sa firma snaží rozvíjať v rámci dodávateľov tým, že sa zameriava na menšie až stredné farmy, ktoré im môžu dodávať svoje výrobky, ktoré sú následne predávané na internetovej platforme Rohlíku, či už prostredníctvom aplikácie alebo internetového prehliadača.

3.3 Ako funguje Rohlík.cz

Nákup potravín prostredníctvom tohto online obchodu je veľmi jednoduchý a užívateľovi stačia základné poznatky o využívaní počítača alebo mobilného telefónu na to, aby svoj nákup úspešne objednal a dostal ho až k dverám svojho domu.

Nákup je možné vykonať bez toho, aby bol užívateľ registrovaný, pričom registrácia ponúka rôzne výhody, ako zapamätanie predošlých nákupov, zbieranie bonusových kreditov, či umožňuje zákazníkovi si zakúpiť službu *Premium*, ktorá ponúka niekoľko výhod v rámci nákupu.

Samotný proces nákupu je vykonaný zhruba v tomto slede:

1. Zákazník pridá produkty, ktoré si chce objednať, do svojho virtuálneho košíka
2. Zákazník si zvolí dátum a čas dodania nákupu, pričom je pri každej časovej možnosti zobrazená cena (čím menej času od objednania, tým je donáška drahšia)
3. Zákazník prejde k virtuálnej pokladni, kde zadá svoje kontaktné údaje, vrátane telefónneho čísla a adresy
4. Zákazník zvolí spôsob platby z viacerých možností (kvôli pandémie sa momentálne dá platiť len bezkontaktné prostredníctvom online platby)

Po vytvorení objednávky ostáva zákazníkovi čakať na jej dodanie. Po dodaní objednávky je zákazníkovi poslaný dotazník, v ktorom môže vyjadriť svoje pocity, resp. svoju recenziu k danej objednávke.

3.4 Analýza vonkajších faktorov

Na analýzu vonkajších faktorov bude použitý model SLEPTE, ktorý bol opísaný v predošlej kapitole v časti zameranej na analytické nástroje.

3.4.1 Spoločenské faktory

Dôvod pre vznik samotnej firmy je uvedený na ich stránke, z ktorých viackrát spomínaný je práve dôvod eliminácie čakania v dlhých radoch pri pokladni. [9]

Vzhľadom na to, že hlavná myšlienka služby, ktorú poskytuje firma Rohlík, je dovoz základných potravín až k dverám domovov ich zákazníkov bez žiadneho vyčkávania v radoch, je možné predpokladať, že cieľovou skupinou sú ľudia, ktorí neradi čakajú v rade alebo navštevujú obchody osobne. [9]

Z hľadiska dnešnej situácie určite ide aj o ľudí, ktorí sa boja riskovať potenciálne infikovanie vírusom COVID-19.

Je možné, že ak pandémie skončí, budú zákazníci chcieť vystriedať online nakupovanie za to originálne, fyzické nakupovanie, ktoré ich núti ísť medzi iných ľudí, dôvodom čoho by bol nedostatok sociálneho kontaktu vzhľadom na opatrenia, ktoré boli počas posledného roka zavedené.

3.4.2 Technologické faktory a ich vývoj

Z hľadiska technológií ťaží podnik najmä z vývoja mobilných a komunikačných technológií.

Podnik ráta s tým, že dnes už takmer každý vlastní či už osobný počítač alebo mobilný telefón. Prostredníctvom oboch platforiem je možné objednať si nákup a využívať služby daného podniku.

Ich aplikáciu s názvom „Rohlík.cz“ je možné stiahnuť na Android prostredníctvom Google PlayStore [11] alebo na iOS prostredníctvom App Store [12].

Výhody objednávok z domova zahŕňajú aj využitie zabezpečenia s pomocou platobnej brány, prostredníctvom ktorej je možné zaplatiť za službu platobnou kartou cez internet.

Cez pandémiu je možné za službu platiť len platobnou kartou dopredu, a to kvôli tomu, aby sa spoločnosť vyvarovala potenciálnemu šíreniu už spomínaného vírusu. [5]

3.4.3 Ekonomické faktory

V roku 2020 pandémie zapríčinená vírusom COVID-19 ovplyvnila aj český trh a jeho ekonomiu.

Okrem zvýšenej miery inflácie z 2,8% (2019) na 3,2% (2020), vzrástli o 3,2% spotrebiteľské ceny, pričom najväčší vplyv na zvýšenie tejto hladiny mali práve potravinové výrobky, alkoholické nápoje a iné, teda produkty, ktoré tvoria hlavnú časť predaja analyzovaného podniku. [14]

Okrem toho sa v Q4 roku 2020 všeobecne znížili celkové tržby maloobchodov o 7%. [14]

Vo svojej poslednej výročnej správe firma Velká Pecka s.r.o. (známejšie Rohlík.cz) konštatuje, že práve pandémie priniesla ich spoločnosti veľký dopyt po ich službách, čo paradoxne znamenalo vysoký nárast ich predajov, najmä v oblasti potravín. [8]

Z hľadiska všeobecnej situácie krajiny, v ktorej podnik operuje, je v krajine udržiavaná tzv. menová politika, pomocou ktorej sa Česká Národná Banka snaží udržiavať celkovú cenovú stabilitu.

ČNB na svojich stránkach uvádza niekoľko prognóz, vrátane potenciálneho zníženia miery inflácie v roku 2022, kedy by svojou politikou chceli dosiahnuť zmenšenie o zhruba 1% (z 3,2% na 2,3%), čím by sa ekonomika mohla zlepšiť.

3.4.4 Politické faktory

Rohlík.cz operuje pod inými názvami vo viacerých krajinách, avšak vyvíjaná aplikácia bude lokalizovaná do českého jazyka a bude sa momentálne týkať len českých pobočiek. Česká Republika je demokratický štát s voľným trhom podporujúci rozvoj súkromných podnikov.

Krajina sa posledný rok nachádza, kvôli pandémie spôsobenej vírusom COVID-19, v núdzovom stave. Tento stav dáva vláde priestor na vytvorenie nových zákonov v rýchlom časovom slede. Šírenie vírusu donútilo vlády po celom svete sprísniť opatrenia vo svojich krajinách, výnimkou čoho nebolo ani Česko.

Medzi sprísnenia patrili najmä obmedzenia pohybu, dôsledkom čoho bolo skrachovanie malých podnikov a zvýšenie nezamestnanosti. Paradoxne, tieto sprísnenia donútili ľudí hľadať alternatívy, medzi ktoré patria aj online potraviny. Práve posledný rok priniesol firme veľké zisky a zvýšený dopyt ich donútil rozšíriť svoje skladové priestory.

3.4.5 Právne faktory

Ako predajca musí firma disponovať fakturačným systémom, ktorým môže faktúrovať svojich zákazníkov pri kúpe tovaru. Z tohto hľadiska by na svojich stránkach mali mať taktiež uvedené postupy na reklamáciu ich služby a obchodné podmienky.

Keďže pri registrácii užívateľov poskytuje každý z nich svoje osobné údaje, ktoré sú potrebné na dodanie ich objednávok (telefónne číslo, adresa, meno a priezvisko, platobné informácie, atď.), musí firma dodržiavať pravidlá predpísané GDPR, teda zákon o ochrane osobných údajov, ktorý je čoraz prísnejší. Vzhľadom na to musí mať stránka uvedené informácie o spracovaní osobných údajov, aby zákazník vedel, ako bude firma s jeho informáciami zaobchádzať.

3.5 Porterova analýza

V tejto časti bude pomocou Porterovej analýzy opísaná konkurencia analyzovaného online obchodu s potravinami.

Vzhľadom na to, že Rohlík.cz nie sú jediný podnik ponúkajúci túto službu vo svojej lokalite, môže byť perspektíva z hľadiska konkurencieschopnosti nápomocná pri rozpracovaní ďalších metód v tejto práci. Dôvodom analýzy je nájsť oblasti, v ktorých by firma mohla firma inovovať svoje procesy, a tým pádom predbehnúť konkurenciu, ktorá je v Českej republike značná.

3.5.1 Hrozba súčasnej konkurencie

Súčasná konkurencia zahŕňa tri ďalšie online potraviny, poskytujúce v podstate rovnaké služby. Týmto podnikmi sú konkrétne iTesco, Košík a iGlobus.

Vzhľadom na počet konkurentov nejde práve o presýtený trh, v súčasnosti je v podstate možné si vybrať len z týchto štyroch služieb (vrátane Rohlíku), pričom rozhodujúca môže byť najmä lokalita zákazníka a prístup, ktorý má podnik k zákazníkovi v prípade nejakého problému.

Okrem lokality môžu byť rozhodujúce taktiež ceny, avšak zdá sa, že tie medzi jednotlivými obchodmi nie sú až tak odlišné natoľko, aby si zákazník vyberal práve na základe ich výšky. Samotné špecifikácie všetkých týchto podnikov sú veľmi podobné, a tým pádom sa dá skonštatovať, že v konečnom dôsledku bude okrem lokality záležať výber poskytovateľa hlavne od preferencie zákazníka.

V roku 2018 analyzoval tieto obchody (mimo iGlobusu) internetový portál Refresher, kde podrobne rozoberali viacero aspektov jednotlivých subjektov, vrátane cien, kvality stránok a úroveň zákazníckej podpory. [22] Magazín VímVíc nedávno uskutočnil podobnú, avšak menej detailnú analýzu týchto online obchodov. [23]

V oboch prípadoch bol na prvom mieste práve online „refazec“ Rohlík, ktorý si toto miesto vyslúžil najmä svojím postojom k zákazníkovi, čím v podstate spĺňajú svoju misiu, ktorú si sami nastavili.

3.5.2 Hrozba nových firiem v odvetví

V tomto smere záleží najmä na fakte, či by chcel konkurent do tohto odvetvia vstúpiť s úplne novou firmou a budovať ju od základov, alebo by to bola adaptácia už existujúceho kamenného obchodu/reťazca, ako napríklad iTesco.

Podľa prieskumov nakupuje potraviny online aspoň raz mesačne zhruba 16% Čechov, pričom momentálna kríza spôsobená nútenou karanténou by mohla toto percento zdvihnúť o niečo viac. Z hľadiska obchodov ide o pomerne novú službu, ktorá sa začala zhruba v polke minulého desaťročia, čo by znamenalo, že ešte stále je ideálny čas pre vstup nových konkurentov do tohto odvetvia.

Podľa článku z VímVíc je vraj niekoľko známych mien v rámci obchodov potravín v strede jednaní o založení vlastných online služieb.

Lidl momentálne neumožňuje kúpu potravín online, avšak integrácia z hľadiska webového portálu a aplikácie implementovaná už je, čo by mohlo samotný prechod na iný druh tovaru výrazne urýchliť.

Na trh nedávno vstúpila na tento trh firma Globus ČR, s.r.o. so svojou online službou iGlobus. Rozšíreniu podniku do sveta internetového sprostredkovania prispela súčasná situácia. V prípade tohto obchodu ide, podobne ako v prípade Tesca, taktiež o rozšírenie platformy, keďže samotné pobočky Globusu môžu byť nájdené na viacerých miestach v Českej Republike.[35]

3.5.3 Hrozba rastúcej kompetencie dodávateľov

Rohlík.cz je samozrejme závislý na dodávateľoch a ich produktoch. Z hľadiska kompetencie vie byť pri väčšine dodávaných tovaroch podnik flexibilný, keďže na trhu môžu nájsť viacero substitútov (najmä ak ide o bežné potraviny). V tomto prípade je možné dôjsť k záveru, že firma nie je tak závislá na jednom dodávateľovi, takže nemusí čakať veľkú hrozbu z tohto hľadiska.

Dalo by sa tvrdiť, že sa takému vývoju snažia sami zabrániť, a to kvôli ich vlastnej iniciatíve, ktorá má privítať malých dodávateľov. V roku 2018 totiž firma založila v spolupráci s obchodom Sklizeno investičnú skupinu MFC, ktorá má pomôcť malým až stredným farmám dostať ich produkty práve na online platformu poskytovanú analyzovaným podnikom. O samotnú kvalitu dodávaného tovaru sa stará Sklizeno. [3]

V súčasnej situácii im toto hnutie zabezpečuje prísun tovaru. V dôsledku rôznych zmien, vrátane občasného zákazu vychádzania, môžu malí až strední dodávatelia predávať svoje domáce produkty prostredníctvom internetu, pričom je možné predpokladať, že drvivá väčšina z nich by nevedela osamote vyžiť, poprípade sa prispôbiť podmienkam, ktoré zvýhodňujú práve donášky potravín a jedál až k dverám zákazníkov.

3.5.4 Hrozba rastúcej kompetencie zákazníkov

Dá sa konštatovať, že vyjednávacía sila zákazníkov sa zvyšuje s počtom konkurentov na trhu. Keďže Rohlík.cz nie je jediným online obchodom s potravinami na trhu, musí o zákazníkov bojovať svojou kvalitnou službou. Keďže celý proces dodávania tovaru zákazníkom začína v ich skladoch, je nutné zaistiť čo najlepšie podmienky v týchto priestoroch, a to najmä pre zamestnancov, ktorí v nich pracujú.

Podnik sa snaží mimo kvality svojich produktov udržiavať a zvyšovať svoju kvalitu najmä po stránke zákazníckej podpory. Ich linky sú otvorené nielen v pracovné dni, ale aj cez víkend. V prípade reklamácie služby sa podnik snaží daný nedostatok čo najskôr odčiniť, či už prostredníctvom vrátenia peňazí alebo iných prostriedkov.

3.5.5 Hrozba substitučných produktov

Z hľadiska substitučných produktov, ako už bolo spomenuté, existuje viacero konkurentov, ktorý poskytujú rovnaké služby a s nimi späté produkty. Možnou výhodou Rohlíku môže byť ich podpora malých fariem a dodávateľov, ktorí by im mohli dodávať viacero výrobkov, tým pádom rozšíriť ich celkový sortiment. To by mohlo prilákať viacero zákazníkov od konkurencie. Rohlík.cz však musí taktiež dbať na kvalitu dodávania a výšku cien.

3.6 Analýza vnútorných faktorov

Analýza bude vykonaná pomocou metódy 7S od McKinseyho.

Systémy

Momentálne sa na komunikáciu vo firme používajú nasledujúce aplikácie:

- **Trello.** Ide o systém agilného rozhrania, kde je užívateľom poskytnutá veľká tabuľa, na ktorú môžu virtuálne pripínať karty. Tieto karty môžu byť kategorizované rôznym spôsobom značkami, ktoré si administrátor definuje. Firma túto stránku používa práve na vytváranie úloh pre svojich zamestnancov, ktorí sa na tieto úlohy prostredníctvom Trelly prihlásia, resp. si tú kartu vezmú na starosť. Práve táto aplikácia je subjektom zmeny v skladových priestoroch, keďže firma potrebuje špecifickejšie nastavenia svojich kariet.
- **Slack.** Komunikačná aplikácia určená najmä pre podniky. Ponúka klasické funkcionality, ako posielanie správ a skupinové volanie, či už videohovory alebo čisto hlasové telefonáty. Ponúka rôzne integrácie pre iné aplikácie, napr. Google Docs, GitHub, atď.

Na klasické, kancelárske práce, typu vytváranie dokumentov, prezentácií a tabuliek sa používa balíček Microsoft Office 365.

- MS Word - využívaný na bežné vytváranie dokumentov
- MS Excel - vytváranie rôznych tabuliek potrebných k uchovávaní dát
- MS PowerPoint - softvér na vytváranie prezentácií v rámci firmy pre rôzne stretnutia projektových tímov

Stratégia

V podstránke „Naše hodnoty“ stanovuje Rohlík.cz hneď ako prvé heslo „*Customer is in the centre of our universe.*“, čo znamená „*Zákazník je stred nášho vesmíru.*“ To veľmi jednoducho popisuje cieľ ich služby. Misiou podniku je zaručiť zákazníkovi čo najlepšiu službu v rámci doručenia potravín až k jeho dverám. Podľa ich stránok je zákazník na prvom mieste, a urobí všetko pre to, aby mu čo najviac vyhovel. Ich heslom je „*Better done than*

perfect.“, teda „*Dokončená činnosť je lepšia než perfektná.*“. [9] Dá sa povedať, že heslo ide ruka v ruku s myšlienkou toho, že je lepšie zbierať skúsenosti a poznatky a vylepšovať ich službu za cesty.

Z hľadiska zamestnania a pracovnej morálky sa snaží firma udržiavať atmosféru, ktorá je dnes bežná pre start-up firmy. Ide o priateľské prostredie s dobrou náladou, rôznymi výhodami a podporou osobného rozvíjania.

Štruktúra

Z hľadiska štruktúry sa podnik delí na tri rôzne oblasti, a to *logistika*, *sklad* a *centrála*. [9]

Prvé menované má na starosti rozvážanie potravín, takže zahŕňa kuriérov, dispečerov a správcov vozového parku. Podľa ich stránok je v ČR celkovo zamestnaných cez 500 kuriérov a dokopy obsluhujú zhruba 200 kuriérskych áut. [9]

Sklad zahŕňa skladníkov, pekárov a koordinátorov skladu. Podľa stránky sa napríklad v skladoch v Prahe denne roznáša zhruba 6 000 objednávok, pričom každú hodinu to tvorí zhruba 28 000 kusov tovaru. Ako bolo spomenuté, sklad zahŕňa aj pekárov, keďže Rohlík sa pýši vlastným, čerstvo vyrobeným pečivom. [9]

Posledná sekcia je centrála. Tá v sebe nosí niekoľko oddelení, ktoré dokopy vytvárajú infraštruktúru zabezpečujúcu chod ich aplikácie a produktu.

- Finančné oddelenie
- Komerčné oddelenie
- Zákaznícka podpora
- IT oddelenie
- Business intelligence
- Strategické oddelenie
- Produktové oddelenie
- Marketing
- Human Resources

Štýl riadenia

Vzhľadom na to, že Rohlík je celkom mladá firma, a jej hlavným produktom je v podstate ich aplikácia a web stránka na objednávanie potravín, ide o podnik, ktorý sa snaží chovať moderne. Ich zamestnanci sú odmeňovaní rôznymi výhodami, ako napr. multisport karta, objednávky nákupov s dopravou zdarma, tímové aktivity, či home office, ktoré sú najmä v dobe pandémie veľmi žiadané.

Mimo toho, ako už bolo spomenuté, firma presadzuje filozofiu, v ktorej by sa zamestnanci nemali báť prejavíť svoj vlastný názor. Práve naopak, hľadajú ľudí, ktorí vedia byť automaticky prirodzenými lídrami v rámci firmy, čo začína byť štandardom v podobne nových firmách. Svojim zamestnancom práve ponúkajú kariérny rast priamo v ich firme. Hoci ide o pomerne nový podnik, už dnes je tam dostatok zamestnancov a vyšších pozícií na to, aby ľudia mohli postupovať vyššie a získavať potrebné skúsenosti na ďalší postup v budúcnosti a v ich kariérnom živote.

Rohlík pre zamestnancov ponúka aj knižnicu, v ktorej sa ich pracovníci môžu vzdelávať. Rovnako sa vzdelávať môžu aj prostredníctvom rôznych jazykových kurzov a iným školením.

Mnoho týchto výhod práve motivuje pracovníkov sa celkovo zlepšovať, čím sa zvyšuje aj celková morálka.

Spolupracovníci

Ako bolo spomenuté, firma v ČR zamestnáva podľa poslednej ročnej správy zhruba 300 ľudí. S rozšírenými skladmi a infraštruktúrou sa očakáva, že sa toto číslo takmer zdvojnásobí.

Na postoch kuriérov a dodávateľov bývajú zamestnaní študenti, ktorí si chcú privyrobiť popri škole. Mimo toho sa firma snaží zamestnávať skúsených ľudí, bez ohľadu na vek. Vzhľadom na náтуру firmy je však viac lákavá práve pre mladšie kategórie, ktorých môžu lákať už práve spomenuté výhody, či mladý kolektív a kamarátska atmosféra, v ktorej sa preferuje tykanie pred vykaním.

Jeden z tých zaujímavejších benefitov môže byť využitie aplikácie Bonusly, ktorá dáva zamestnancom možnosť hodnotiť sa medzi sebou. Čím lepšie hodnotenie zamestnanec od druhých dostáva, tým väčšie výhody si v aplikácii Bonusly môže dovoliť. Medzi benefity patria napríklad masáže, poukážky do rôznych obchodov, atď. Tento prístup môže nielen motivovať zamestnancov k lepšej práci, ale taktiež k lepšej spolupráci v tímoch.

Zdieľané hodnoty

Medzi najdôležitejšie hodnoty firmy patria vynaliezavosť, profesionalita, tvorivosť a zodpovednosť. V predošlých bodoch bola spomenutá aj schopnosť robiť vlastné rozhodnutia a prinášať svoje individuálne sily do tímovej práce.

V Rohlíku sa všetci snažia pracovať ako jeden, keďže celý proces a infraštruktúra zahŕňa v podstate každého pracovníka, od IT oddelenia až po kuriéra. Podnik sa snaží udržiavať tieto hodnoty prostredníctvom rôznych tímových aktivít, ktoré prebiehajú mimo pracovného času, či už ide o klasický *team-building* alebo nejaký večierok, ako oslava narodenín jedného zo zamestnancov.

Schopnosti

Rohlík sa v poslednom roku vypracoval na jednu z najlepších firiem v Európe z hľadiska adaptácie na nové podmienky vytvorené pandémiou Európe. Za rok 2020 totiž získali zlatú cenu v prestížnej konferencii ECCCSA v novej kategórii „*Responding in a Crisis - Best Partnership Solution (Outsourcing)*“. [10]

Kategória sa zamerala na firmy, ktoré sa vedeli najlepšie prispôbiť novým podmienkam na trhu a v práci, teda v rámci chodu firmy. Cena odmieňa tímovú prácu v ťažkých časoch. [10]

3.7 Zhrnutie analýz - SWOT

V tejto časti budú celkovo zhrnuté doteraz zistené poznatky pomocou predošlých rozborov, ktoré boli spravené pomocou analýzy SLEPT, Porterovho modelu a modelu 7S. Zhrnutie bude vykonané populárnou analýzou *SWOT*, teda silných a slabých stránok, príležitostí a hrozieb.

Silné stránky

Silnou stránkou firmy je jej prístup k zákazníkom. Tým, že ich pokladajú za najdôležitejších, stále zdokonaľujú svoj produkt a svoju službu. Po vypuknutí vírusu SARS-CoV-2 sa okamžite prispôbili podmienkam, ktoré uložila vláda. Rýchla mobilizácia pracovníkov a vytvorenie nových skladov ukázalo, že sa dokážu efektívne adaptovať na vonkajšie faktory.

Ďalšou silou podniku je skvelá súdržnosť zamestnancov, ktorí sú motivovaní rôznymi tímovými aktivitami v rámci, ale aj mimo firmy. Celá firma má vlastný Slack kanál, na ktorom môže každý zamestnanec komunikovať s inými.

Expanzia je ďalším dobrým bodom pre podnik, keďže využili situáciu nepokrytého trhu v blízkych krajinách Európskej Únie. To im otvára príležitosti expandovať, v prípade úspechu, ďalej v týchto štátoch.

Slabé stránky

Medzi slabé stránky patrí práve chýbajúca integrácia nejakého špecifického systému, kde by sa mohli zapisovať záznamy, teda úlohy. Kvôli novým skladom v Rohlíku pribudlo viac práce, a tým pádom je takýto systém, resp. softvér potrebný. Podnik momentálne používa Trello, ale je niekoľko vecí, ktoré vedeniu firmy v tejto aplikácii chýba, ako napr. skenovanie lokácie prostredníctvom QR kódu alebo jednoduchšie rozhranie.

Vo firme je s rozširujúcou sa infraštruktúrou potrebné väčšie IT oddelenie. Momentálne toto oddelenie zamestnáva na niektorých pobočkách len pár zamestnancov, pričom na inštaláciu, poprípade opravu počítačovej techniky, je potrebných viacero kvalifikovaných ľudí, ktorými momentálne podnik nedisponuje v ideálnej miere.

Príležitosti

Rohlík už má skúsenosť s expanziou svojej služby do ďalších krajín, keďže svoje pobočky majú už aj v Maďarsku a Rakúsku. Práve skúsenosť a súčasná situácia, v ktorej sa stále vyžaduje služba online potravín, odкрýva vedeniu niekoľko príležitostí, ktorých by sa mohli ujať. Vzhľadom na priebeh pandémie, ktorá ustupuje veľmi pomalým tempom, sa firme otvára možnosť rozšírenia do ďalších miest v spomenutých krajinách, poprípade iných krajín.

Medzi príležitosťami taktiež možno započítať predaj nových druhov výrobkov. Rohlík už teraz presadzuje iniciatívu zakomponovania malých až stredných dodávateľov vo svojom výbere produktov. Momentálne sa jedná o farmy, ale postupom času by firma mohla dať možnosť aj iným druhom výrobcov, ktorí sú v dnešnej dobe na pokraji krachu kvôli nedostatku zákazníkov dôsledkom opatrení, ktoré zakazujú ľuďom navštevovať maloobchody.

Hrozby

Jedna z hrozieb môže byť náhly pokles zákazníkov. Dá sa predpokladať, že po skončení pandémie a návrate do normálu bude viacero ľudí znovu preferovať fyzické nakupovanie, minimálne na nejakú krátku dobu, keďže sa konečne bude dať po dlhšej dobe pohybovať voľne.

S rozšírenou službou v iných krajinách prichádza risk. Podniku sa v iných krajinách nemusí dariť a môže sa stať, že pobočky nebudú v krajinách mimo Českej republiky také ziskové, ako sú v nej.

Ďalšou hrozbou môže byť otvorenie viacerých online služieb iných, známych a osvedčených veľkoobchodov. V tom prípade by mohlo byť pre Rohlík ťažšie získať zákazníkov. Možným konkurentom sú obchody ako Lidl, ktorý napríklad už online službu má, avšak zatiaľ v Českej Republike predáva len domáce potreby a oblečenie, resp. nepredáva potraviny, ako Rohlík alebo iTesco. S už implementovanou stránkou by však mohli mať našliapnuté na rozšírenie služby o potraviny.

Silné stránky	Slabé stránky
<ul style="list-style-type: none"> • súdržnosť • tímový duch • odviazaná atmosféra • definovaná vízia a misia • prispôsobenie sa na nové podmienky • skúsené vedenie • finančná situácia 	<ul style="list-style-type: none"> • neorganizovaná údržba v sklade • vyššie ceny ako v kamenných obchodoch • nedostatočný počet kuriérov vzhľadom na počet objednávok • objednávky nedodávané na čas • nízky počet kvalifikovaných zamestnancov v IT oddelení
Príležitosti	Hrozby
<ul style="list-style-type: none"> • expanzia na ďalšie trhy • nové druhy výrobkov • využitie pandémie na získanie viacerých zákazníkov 	<ul style="list-style-type: none"> • prudký pokles zákazníkov po návrate voľného pohybu • nezaplnenie pozícií kuriéra a skladníka • stagnácia na cudzích trhoch • skolabovanie informačných systémov a ICT

Tabuľka 3.1: Tabuľka SWOT (zdroj: vlastná tvorba)

Kapitola 4

Vlastné návrhy riešenia, prínos návrhov riešení

Náplňou tejto kapitoly je predstavenie návrhu aplikácie a následný opis celkového procesu implementácie do podniku. Súčasťou kapitoly bude aj krátka dokumentácia aplikácie a opis funkcionalít. V závere tejto časti sa vyhodnotí úspešnosť navrhovaného riešenia.

4.1 Aktuálny stav z hľadiska implementácie

Predtým, než bude predstavený samotný návrh zmeny, je potrebné uviesť problém, ktorý má navrhované riešenie eliminovať.

V súčasnosti sa v Rohlíku kvôli väčšiemu dopytu rozširujú skladové priestory, čo znamená väčší počet zamestnancov na udržiavanie väčšej plochy, viac prístrojov a techniky na lepšie zaobchádzanie s tovarom, poprípade iným majetkom nachádzajúcim sa v týchto priestoroch. Okrem iného ide samozrejme aj o vecí týkajúce sa skladovaných potravín a iného tovaru. V konečnom dôsledku, so zväčšujúcim sa úložným priestorom sa taktiež zvyšuje celkový počet potenciálnych problémov, ktoré sa môžu vyskytnúť v rámci udržiavania chodu skladu. Faktom je, že podnik momentálne nedisponuje žiadnou technológiou, ktorá by im umožnila rýchlejšie komunikovať ohľadom nedostatkov, ktoré sa vyskytnú. Momentálne ide skôr o individuálne riešenia, či už prostredníctvom online kanálu Slack alebo zapisovanie do agilného systému Trello.

4.2 Požiadavky na nový softvér

Správcovia skladu sú nespokojní s momentálnym riešením z hľadiska zaznamenávania a hlásenia chýb. Podnik nedisponuje dostatočne rýchlym a jednoduchým riešením, ktoré je ľahké na pochopenie a rýchle na väčšine zariadení. Ako už bolo spomínané, momentálne riešenie vie byť občas veľmi pomalé, a to najmä na slabších zariadeniach, či už počítačových alebo mobilných.

Faktom je, že softvér Trello môže byť náročný na staršie modely mobilných telefónov. Zátťaž na hardvér robia najmä animácie tejto aplikácie. Ďalším problémom je samotná cena prémiovej verzie danej služby. Cena na mesiac na jedného užívateľa je 12 dolárov. Ak vezmeme do úvahy, že v sklade je zamestnaných 100 užívateľov, môže ísť o výdaj v hodnote 1200 dolárov mesačne za to, aby firma mohla využívať pár funkcionalít z prémiového balíčka. Presné porovnanie, vrátane cenového rozdielu, týchto dvoch riešení bude rozobraté v jednej z ďalších podkapitol.

Ďalšou požiadavkou je rýchle filtrovanie kariet podľa lokality. Momentálne je pre administrátorov skladu ťažké rozlíšiť, v ktorej sekcii skladu sa daný problém nachádza. So zväčšujúcimi sa skladmi ide o čoraz náročnejšiu orientáciu v týchto priestoroch, preto je potrebné pre firmu mať okamžite dostupné informácie o tom, kde treba ísť niečo opraviť, poprípade doplniť tovar, atď.

Poslednou je vlastná databáza, pomocou ktorej vie podnik štatisticky určovať najčastejšie chyby. To by im dávalo lepší prehľad nad tým, čo sa deje v rámci ich skladov. Manažovanie vlastných back-end technológií a databázy im totiž dáva úplne voľnú ruku vzhľadom na zber dát a ich udržiavanie.

4.3 Predstavenie nového riešenia

Nové riešenie má za úlohu implementovať systém, ktorý by umožňoval zamestnancom jednoducho a rýchlo vytvárať „tikety“, ktoré by v sebe niesli informácie, ako napr. miesto problému, opis problému, autor, dátum, prioritu, atď. Toto riešenie by bolo možné využívať prostredníctvom mobilnej aplikácie, ktorá by bola prístupná na najrozšírenejšie operačné systémy (Android a iOS). Táto aplikácia bude naprogramovaná pomocou jazyka JavaScript a jeho frameworku React Native. Front-end aplikácie bude naprogramovaný externým vývojárom a back-end (databáza) bude poskytnutá interne, vyvíjaná technickým oddelením podniku.

Obyčajní zamestnanci by mali prístup len k vytváraniu týchto záznamov, pričom vedenie (alebo iní poverení s vyššou právomocou) by malo špeciálne vytvorené rozhranie, ktoré by im umožnilo prístup k týmto záznamom, pomocou ktorých by mohli tieto zaznamenané problémy delegovať pracovníkom, ktorí by daný problém overili, popřípade vyriešili.

Na druhú stranu, administrátori budú mať prístup k väčšiemu rozhraniu. Každý z nich bude mať vlastný účet s väčšími právami, ktoré im dovoľia vidieť všetky tikety v databáze. Títo užívatelia môžu záznamy mazať, spravovať a vytvárať. To im dovoľí mať lepšiu kontrolu a detailnejší prehľad nad problémami v sklade, kontaktovať dotýčnych ľudí, delegovať problémy medzi zamestnancov, atď. Okrem toho by sa výrazne zlepšila celková komunikácia, najmä po rýchlostnej stránke.

4.4 Lewinov model

4.4.1 Fáza rozmrazenia

V tejto fáze budú uvedené sily pôsobiace na navrhovanú zmenu, ich analýza a následné vyhodnotenie, či je nový návrh podporovaný.

V rámci firmy má na danú zmenu vplyv niekoľko faktorov. Pre túto prácu bola zvolená škála síl -10 až +10 (čím väčšie číslo, tým viac daná sila zmenu podporuje).

Sily podporujúce zmenu

Tieto sily chcú zvýšiť celkovú efektivitu procesu, ktorý má za úlohu identifikovať nedostatky v skladoch, ich zaznamenanie a následne ich elimináciu.

V prvom rade ide o technické oddelenie, ktoré má na starosti sklady a ich infraštruktúru. Vedenie tohto oddelenia tlačí na požadovanú zmenu, keďže by v konečnom dôsledku ušetrila čas, a tým pádom aj peniaze.

Hodnota sily: **+10**

Ďalšími advokátmi za zmenu sú samotní zamestnanci, ktorí nie sú spokojní s momentálnym zaznamenávaním problémov, keďže systém nie je momentálne veľmi jednotný v rámci všetkých pobočiek (napríklad v Brne je využívaný iný systém ako v Prahe).

Hodnota sily: **+5**

Sily proti zmene

Z hľadiska síl proti zmene sú tu v podstate dve sily, ktoré by mohli zabrániť či už uskutočneniu procesu alebo jeho zabrzdzeniu.

Jedna z nich sú zamestnanci, ktorým nové riešenie nevyhovuje a nesúhlasia so zavádzaním tohto systému, keďže budú musieť byť školení na využívanie tejto novej technológie. Treba brať do úvahy, že nie všetci zamestnanci skladu sú totiž technicky zdatní, keďže ich práca z väčšiny nezahŕňa práve tieto vedomosti.

Hodnota sily: **-3**

Proti zmene pôsobia skôr vonkajšie faktory, ako súčasná pandémia a jej dopady na zvýšený dopyt, čo znamená, že existujú dôležitejšie medzery v rámci firmy, ktoré je potrebné vyplniť, tým pádom neostáva toľko času na implementáciu novonavrhovanej aplikácie, poprípade vyhradenie prostriedkov na naprogramovanie nového softvéru.

Hodnota sily: **-5**

Sila	Hodnota
Technické oddelenie	+10
Zamestnanci skladu	+5
Zamestnanci odporujúci technológii	-3
Vonkajšie faktory	-5
Výsledná sila	+7

Tabuľka 4.1: Tabuľka síl pre Lewinov model (zdroj: vlastná tvorba)

Po sčítaní pôsobenia rôznych síl je možné konštatovať, že navrhovaná zmena je v rámci firmy podporovaná a je v súlade s plánom rozvoja technológií a skladových priestorov, čo znamená, že zmenu je možné realizovať. Treba podotknúť, že počas vývoja aplikácie sa sila vonkajších faktorov môže zmeniť, poprípade sa môže objaviť nový faktor. Tieto faktory však môžu skôr byť označované ako riziká, proti ktorým budú navrhnuté opatrenia neskôr v tejto kapitole.

Agent zmeny

Agentom zmeny je niekto, kto danú zmenu uskutočňuje. V tomto konkrétnom príklade pôjde o externého vývojára, ktorý bude mať na starosti front-end, respektíve vzhľadovú stránku veci a vývojár priamo z firmy Rohlík, ktorý bude mať na starosti back-end, teda databázu, do ktorej sa budú ukladať jednotlivé záznamy problémov. Tieto záznamy budú vytvárané pomocou front-endu.

Vývoj softvéru je samozrejme dôležité komunikovať s vedením technického oddelenia v Rohlíku, ktoré sa tým pádom tiež dajú považovať za jedného z agentov zmeny. Vedúci oddelenia v tomto prípade bude konzultantom počas projektu.

Sponzorom zmeny je samotné vedenie Rohlíku, konkrétne ich IT oddelenie v spolupráci s finančným oddelením.

Intervenčné oblasti zmeny

- **Ludské zdroje a ich riadenie.** Pomocou zavedenia nového softvéru budú zamestnanci môcť byť efektívnejšie riadení. V prípade problému už nebudú musieť zamestnanci volať svojim manažérom, keďže budú môcť nahlásiť problémy pomocou novej aplikácie.
- **Organizačná štruktúra firmy.** Zmena v tomto prípade nebude meniť organizačnú štruktúru firmy.
- **Technológia firmy.** Zmena zavedie do firmy úplne novú aplikáciu, ktorá bude využívaná zamestnancami, najmä v skladových priestoroch. Je možné, že firma bude musieť zabezpečiť nové technológie, ako sú servery na prevádzku databázy a nové mobilné telefóny, v prípade, že bude telefón zamestnanca nekompatibilný s aplikáciou.
- **Komunikačné a organizačné toky a procesy firmy.** Nová aplikácia urýchli komunikáciu medzi skladníkmi a správcami skladu. Pôjde o zlepšenie procesu v rámci delegovania problémov pomedzi skladových zamestnancov. Taktiež pôjde o rýchlejšie zaznamenávanie a ohlasovanie problémov.

4.4.2 Fáza prechodu a aplikácia zmeny

Predtým, než môže byť samotná zmena aplikovaná, musí byť naprogramovaná a otestovaná. Po následnom testovaní a schválení funkcionalít môže byť nasadená do praxe.

Bližšie detaily budú predstavené v časti popisujúcej priebeh projektu a naplánované sprinty.

4.4.3 Fáza zmrazenia

V tejto fáze bude implementovaná zmena analyzovaná a sledovaná. Na získanie spätnej väzby budú slúžiť rôzne nástroje, napr. na to bude slúžiť tlačítko „Nahlásiť chybu“, kde bude možné pre užívateľa skomponovať správu priamo v aplikácii, ktorá bude slúžiť na odovzdanie spätnej väzby vývojárom za ňu zodpovedným. Okrem iného budú raz za kvartál rozoslané dotazníky ohľadom aplikácie, kde môžu zamestnanci vyjadriť svoju prípadnú nespokojnosť so zmenou.

4.4.4 Záver po zavedení zmeny

Po zhodnotení všetkých odpovedí a pripomienok bude nasledovať stretnutie s technickým oddelením Rohlíku, kde bude uskutočnená dohoda na ďalších krokoch v prípade vývoja aplikácie, resp. nových funkcionalít alebo opráv, ktoré by mali byť vytvorené, poprípade uskutočnené na zvýšenie kvality celkového UX (angl. user experience - skúsenosť užívateľa).

Tento krok bude cyklický, keďže pôjde o neustále vylepšovanie aplikácie a aktualizáciu technológií, pomocou ktorých bol softvér vyvinutý.

Zmena sa bude považovať za úspešnú, ak bude softvér implementovaný priamo do chodu skladu a bude využívaný v plnom rozsahu na dennej báze. Vzhľadom na výstavbu nových skladov však úplne zavedenie softvéru do prevádzky a zaškolenie zamestnancov môže trvať nejaký čas.

4.5 Analýza rizík pomocou RIPRAN

V tejto časti bude vykonaná analýza rizík pomocou metódy RIPRAN. Pôjde o identifikáciu hrozieb a ich scenárov, ohodnotenie ich pravdepodobností a dopadov, a následné celkové verbálne ohodnotenie na základe kombinácie pravdepodobností a dopadov.

4.5.1 Určenie sústavy

Pred identifikáciou je nutné určiť sústavu, podľa ktorej budú hrozby analyzované. Na základe množstva informácií a veľkosti projektu bude využitá sústava 3x3x3.

Verbálne ohodnotenie	Pravdepodobnosť v %
Nízka pravdepodobnosť (NP)	Pod 33%
Stredná pravdepodobnosť (SP)	33% až 66%
Vysoká pravdepodobnosť (VP)	Nad 66%

Tabuľka 4.2: Tabuľka stupňov pravdepodobnosti [25]

Na ohodnotenie pravdepodobnosti budú použité stupne veľkej, strednej a malej pravdepodobnosti. Každá z nich označuje rozpätie pravdepodobnosti udané v tabuľke 4.2.

Verbálne ohodnotenie	Význam
Nízky nepriaznivý dopad projektu	<ul style="list-style-type: none">• Škody do 0,5% z celkovej hodnoty projektu• Dopady vyžadujúce určité zásahy do plánu
Stredný nepriaznivý dopad projektu	<ul style="list-style-type: none">• Škoda od 0,51 do 19,5% z hodnoty projektu• Ohrozenie termínu, nákladov

Veľký nepriaznivý dopad projektu	<ul style="list-style-type: none"> • Ohrozenie cieľa projektu • Ohrozenie koncového termínu projektu • Možnosť prekročenia celkového rozpočtu • Škoda cez 20% z hodnoty projektu
----------------------------------	--

Tabuľka 4.3: Tabuľka ohodnotenia dopadu

Na ohodnotenie dopadu budú zas použité tri rozličné stupne, v ktorej každý stupeň znamená nejaký konkrétny dopad na projekt, či už finančný alebo vecný:

	VD	SD	ND
VP	VHR	VHR	SHR
SP	VHR	SHR	NHR
NP	SHR	NHR	NHR

Tabuľka 4.4: Tabuľka výsledných možných hodnôt rizika [25]

- VHR - vysoká hodnota rizika
- SHR - stredná hodnota rizika
- NHR - nízka hodnota rizika

Celkové ohodnotenie je určené na základe "priemeru" medzi pravdepodobnosťou a dopadom rizika. Táto hodnota bude určená pomocou výslednej stupnice v tabuľke 4.4.

4.5.2 Identifikácia a ohodnotenie rizík

Keďže sústava a jej hodnoty boli definované, ďalším krokom je identifikovať riziká a určiť ich pravdepodobnosť a dopady pomocou tejto metódy.

Riziká budú rozdelené do niekoľkých skupín, z ktorých väčšina má nízku pravdepodobnosť, avšak vysoký dopad.

ID	Hrozba	Scenár	P	D	H
1	Zákaz vstupu do skladu kvôli opatreniam	Aplikácia sa nebude dať otestovať.	NP	VD	SH
2	Nedostatočný rozpočet	Neschopnosť platiť vývojára.	NP	VD	SH
3	Únik informácií z databázy	Zlá reklama, strach z úniku osobných údajov zákazníkov.	NP	VD	SH
4	Problémy so schvaľovaním aplikácie	Aplikáciu nebude možné distribuovať medzi zamestnancov.	SP	VD	VH
5	Neochota spolupráce zamestnancov	Aplikácia nebude mať dobré meno medzi zamestnancami v sklade.	NP	VD	SH
6	Zneužívanie aplikácie	Zamestnanci nebudú brať aplikáciu vážne. Zahltenie servera.	NP	SD	NH
7	Výpadok databáze	Aplikácia sa nebude dať používať.	SP	VD	VH
8	Strata všetkých informácií	Záznamy budú úplne nedostupné.	NP	VD	SH
9	Nie každý zamestnanec má iOS alebo Android	Aplikáciu nebudú môcť využívať niektorí zamestnanci.	SP	SD	SH
10	Výpadok siete	Aplikácia sa nebude dať používať.	SP	VD	VH

Tabuľka 4.5: Tabuľka rizík pomocou metódy RIPRAN

Z tabuľky je možné vyčítať, že väčšina z týchto rizík má teda vysokú hodnotu rizika, a tým pádom by mali byť zavedené opatrenia, ktoré by mali buď znížiť pravdepodobnosť hrozby alebo znížiť jej dopad. Cieľom tejto analýzy je práve vymyslieť vhodné opatrenia na minimalizáciu celkovej hodnoty každého rizika.

4.5.3 Výber opatrení a ich vplyv na výsledné hodnoty

Hlavným kritériom bolo minimalizovať všetky riziká, avšak pri niektorých bodoch sa aj napriek zníženiu pravdepodobnosti nepodarilo zmeniť výslednú hodnotu rizika. Ide o to, že v niektorých prípadoch totiž ostáva hodnota dopadu vysoká, čo znamená, že výsledná hodnota je vždy aspoň stredná. Každopádne by sa s týmito opatreniami malo firme podať znížiť celkovú hodnotu rizík, a tým pádom predísť väčšiemu poškodeniu v rámci projektu, či už pri samotnom zavedení alebo pri monitorovaní aplikácie.

ID	Hrozba	Opatrenie	P*	D*	H*
1	Zákaz vstupu do skladu kvôli opatreniam	Vytvorenie umelých podmienok na testovanie.	NP	SD	NH
2	Nedostatočný rozpočet	Lepší dohľad nad financiami a priebežné platenie.	NP	VD	SH
3	Únik informácií z databázy	Lepšie zabezpečenie serveru.	NP	VD	SH
4	Problémy so schvaľovaním aplikácie	Overiť a dodržať podmienky schvaľovania aplikácie.	NP	VD	VH
5	Neochota spolupráce zamestnancov	Zistenie spätnej väzby od zamestnancov pred vývojom.	NP	SD	NH
6	Zneužívanie aplikácie	Penalizácia za zneužívanie a obrana proti spamu.	NP	SD	NH
7	Výpadok databáze	Zabezpečiť záložnú databázu.	NP	SD	NH
8	Strata všetkých informácií	Priebežné zálohovanie dát.	NP	SD	NH
9	Nie každý zamestnanec má iOS alebo Android	Zistenie počtu potrebných telefónov a ich dokúpenia.	NP	ND	NH
10	Výpadok siete	Zabezpečiť mód, ktorý povolí prístup z mobilnej siete.	NP	SD	NH

Tabuľka 4.6: Tabuľka rizík po zavedení opatrení

Vysvetlivky:

- * - označuje hodnoty po opatrení
- **zelená farba** - označuje novú hodnotu po zavedení opatrení

4.6 Časový plán

Predtým, ako bude predstavená analýza nákladov a vlastný návrh riešenia, je ešte nutné predstaviť časový plán, ktorým sa vývojári riadili pri vytváraní tohto softvéru. Na začiatku práce bola opísaný agilný vývoj pomocou sprintov. Práve táto metóda bola použitá pri vývine tohto softvéru.

Časový plán zahŕňa štyri rôzne sprinty, ktoré budú jednotlivo opísané, spolu s činnosťami, ktoré zahŕňali.

V jednotlivých sprintoch budú činnosti rozdelené a ohodnotené počtom *man-day* (*MD*), čo značí 8 hodín ľudskej práce. Takže ak sa niekde $MD = 2$, znamená to, že daná činnosť trvá zhruba 16 hodín. Väčšina činností sa dá robiť súčasne, nuž väčšina z nich bola vykonávaná práve jednou osobou, takže boli vypracované postupne. Keďže ide o dvojtýždňové sprinty, každý z nich bol naplánovaný tak, aby vyšiel na cca 10 MD.

Samozrejme, veľmi zriedkavo ide každý projekt presne podľa plánu. V prípade, že boli niektoré úlohy nedokončené cez daný sprint, boli dokončené v tom ďalšom, poprípade ešte v tom, v ktorom boli plánované, avšak vývojár v tom prípade pracoval nadčas. V prípade, že čas v bloku bol ušetrený, bol buď využitý na zlepšenie aplikácie, poprípade sa začalo rovno robiť na programovacích úlohách z ďalšieho sprintu.

	Sprint 1	Sprint 2	Sprint 3	Sprint 4	Celkový počet
Odhad	10	10	10	10	40
Realita	12,5	8,5	13	10,5	44,5

Tabuľka 4.7: Porovnanie odhadovaných a reálnych počtov MD v sprintoch

4.6.1 Sprint 1

Sprint 1		
Úloha	Odhad MD	Reálne MD
Získanie spätnej väzby od zamestnancov	2	3
Vyhradenie zdrojov	1	1
Návrh aplikácie z hľadiska funkcionality	3	4
Návrh aplikácie z hľadiska dizajnu	3	4
Výber vhodných technológií	1	0,5
Celkový počet MD	10	12,5

Tabuľka 4.8: Sprint 1 - tabuľka činností

V prvom sprinte bolo hlavným cieľom začať celý projekt tým, že bude pripravený návrh aplikácií, a to z hľadiska užívateľského rozhrania a taktiež z pozadia, teda back-end časti aplikácie.

Keďže sa jednalo o novú aplikáciu, ktorú majú používať najmä skladníci, bolo nutné vyhradiť čas na získanie ich spätnej väzby, teda čo by prijali, s čím súhlasia, s čím nie, atď.

Na základe tohto boli vyhradené zdroje a vývojár mohol aplikáciu začať navrhovať z oboch strán, či už z hľadiska funkcionality alebo z hľadiska dizajnu užívateľského rozhrania.

Práve na dizajn bola využívaná aplikácia Figma, prostredníctvom ktorej je možné vyvinúť celkový dizajn mobilného softvéru. Táto stránka umožňuje projektantom zdieľať dizajny nielen medzi sebou, ale aj laikom, ako bolo v tomto prípade vedenie technického oddelenia. To malo prístup k dizajnu počas celého projektu, takže získavanie spätnej väzby fungovalo na dennej báze.

Cieľ tohto sprintu bol splnený mimo časového rozmedzia. Namiesto vyhradených 10 MD to trvalo 12,5 MD, čo však vedeniu podniku nevadilo, keďže im hlavne záležalo na správnom založení projektu.

4.6.2 Sprint 2

Sprint 2		
Úloha	Odhad MD	Reálne MD
Vytvorenie jednotlivých komponentov (front-end)	3	4
Vytvorenie databázy (back-end)	3	2
Prepojenie aplikácie s databázou	1	0,5
Interné testovanie a 1. kolo opráv	3	2
Celkový počet MD	10	8,5

Tabuľka 4.9: Sprint 2 - tabuľka činností

Ide o najdôležitejší sprint, keďže jeho hlavným cieľom bolo naprogramovať prvotnú verziu aplikácie a jej následné otestovanie.

Vytvorenie jednotlivých komponentov do rozhrania trvalo o 1 MD viac, než bolo odhadované, avšak inicializácia databázy a jej samotné prepojenie s rozhraním trvalo kratšie, čo sa odrazilo na samotnom výsledku celého sprintu. Faktom je, že zvolenie správnych technológií (teda MongoDB a Express.js) dovolilo vývojárovi vytvoriť databázu veľmi rýchlo. Keďže sa nejedná o veľmi zložitú aplikáciu, ktorá má v podstate len pár príkazov na back-end, bola práve táto úloha časovo precenená.

S veľkým prekvapením bola teda aplikácia naprogramovaná skôr ako bolo očakávané, keďže bol vývojár schopný ušetriť 1,5 MD, čo využil na zlepšovanie kódu, poprípade štylistické úpravy, ktoré boli doporučené vedením podniku.

4.6.3 Sprint 3

Sprint 3		
Úloha	Odhad MD	Reálne MD
Testovanie v rámci skladu	1	2
2. kolo opráv na základe spätnej väzby	4	5
Príprava business prezentácie	1	1
Tvorba manuálov a videí pre užívateľov	2	3
Dokončenie aplikácie	1	1
Prezentácia vedeniu	0,5	0,5
Registrácia beta verzie (Google Play a App Store)	0,5	0,5
Celkový počet MD	10	13

Tabuľka 4.10: Sprint 3 - tabuľka činností

Tretí sprint mal za úlohu otestovať aplikáciu v praxi, teda v reálnom prostredí skladu. Samotné testovanie aplikácie mal na starosti vývojár, ktorý skúšal niekoľko funkcionáľít, ako blokovanie pripojenia z cudzej siete, scan QR kódov, vytváranie tiketov podľa lokality, atď.

Odhadované dva týždne vývojárovi nestačili na vypracovanie všetkých činností. Najväčším problémom bolo nastavenie sieťových prvkov, teda blokovanie po pripojení z cudzej siete, čo predĺžilo prvú činnosť o 1 MD.

Mimo iného bol taktiež problém s vytváraním tiketov podľa lokality, čo spomalilo druhé kolo opráv a optimalizácií taktiež o 1 MD.

Po samotných opravách nasledovalo konečné dokončenie aplikácie, ktoré priamo nadväzovalo na prezentáciu aplikácie. Keďže prezentácia prebehla úspešne a vedúci technického oddelenia bol spokojný s prácou vývojára, bola dohodnutá distribúcia aplikácie pomedzi zamestnancov skladu po registrácii privátnej beta verzie aplikácie na obchod Google Play Store.

Keďže Apple však nepovoľuje registráciu privátnych aplikácií na App Store, softvér bol už v tomto kroku zaregistrovaný na iOS zariadenia pomocou programu Apple Developer

Enterprise, ktorý dovoľuje firmám distribuovať aplikáciu privátne pomedzi zamestnancov, ktorí majú smartfóny s iOS. Táto služba momentálne stojí 299\$ na rok pre celú firmu.

4.6.4 Sprint 4

Sprint 4		
Úloha	Odhad MD	Reálne MD
Distribúcia aplikácie vybranej skupine skladníkov v Brne	3	2
Analýza a úpravy po spätnej väzbe	6	6
Registrácia plnej verzie (Google Play a App Store)	0.5	0,5
Vydanie aplikácie	1	1

Celkový počet MD	10,5	9,5
------------------	------	-----

Tabuľka 4.11: Sprint 4 - tabuľka činností

Posledný sprint mal za cieľ úplne uzatvoriť vývoj aplikácie, tým pádom aj samotnú implementáciu zmeny do firemného prostredia firmy Rohlík.

Mobilná aplikácia bola distribuovaná vybranej skupine skladníkov a ich správcovi v Brne, pričom na túto činnosť firma vyhradila tri pracovné dni. Vedenie chce priebežne distribuovať aplikáciu po skupinách skladníkov, aby mala aplikácia čas sa dostať do obehu skladu. V tomto čase boli taktiež vylepované QR kódy po sekciách skladu.

Samotné návody boli distribuované pomocou videí a interného PDF súboru, ktorý udáva inštrukcie na prácu s aplikáciou.

Pri tomto kroku nastali menšie komplikácie s inštaláciou aplikácie na niektoré zariadenia, avšak to bolo vyriešené buď objednaním služobného zariadenia alebo inou pomocou pre daný problém.

Posledným krokom celého projektu bolo vydanie aplikácie a jej doterajšie monitorovanie. Podnik sa teda stále snaží aplikáciu postupne podľa svojich potrieb vylepšovať. Vzhľadom na pandémiu je distribúcia aplikácie horšia, ako sa očakávalo. Aplikáciu doteraz využíva zatiaľ jedna skupina zamestnancov, a to najmä kvôli faktu, že so stále zvyšujúcim sa dopytom si firma zatiaľ nemohla kompletne dovoliť prejsť na využívanie úplne nového systému.

4.7 Vlastný návrh softvérovej aplikácie

V tejto podkapitole bude kompletne predstavený návrh riešenia. Pôjde o celkovú prezentáciu aplikácie navrhnuť na implementáciu do firemného prostredia firmy Rohlík. Konkrétne pôjde o pobočku v Brne, ktorá bude v podstate testovacou platformou pre tento nový softvér. Podkapitola bude rozdelená na tri časti: back-end a databáza, front-end a funkcionality aplikácie.

Back-end a databáza

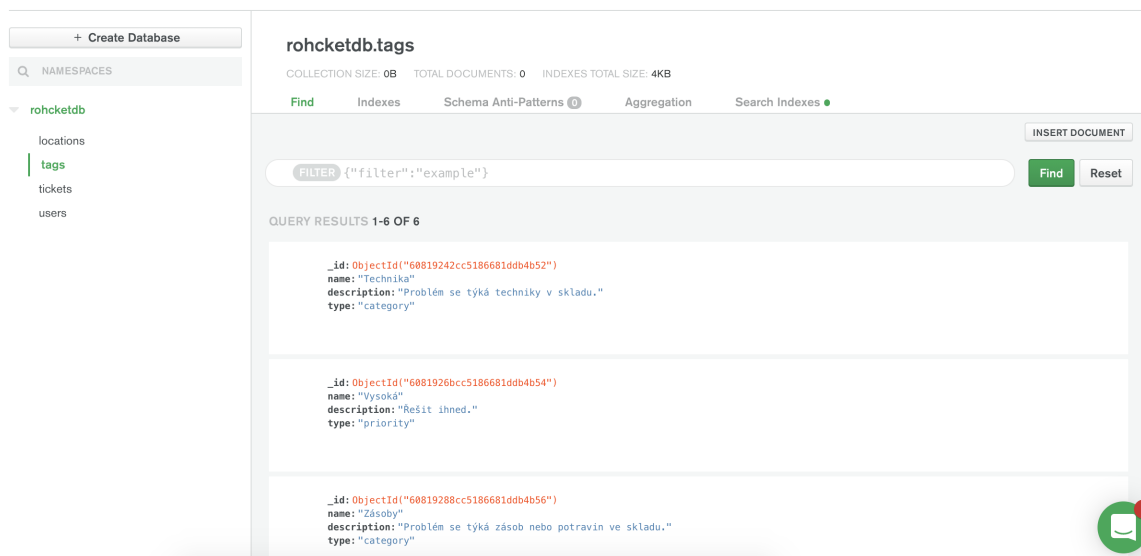
Opis celej aplikácie začne v tejto práci odzadu, teda od serveru a jeho náležitostí. Ako bolo spomenuté na začiatku práce, ide o kombináciu technológií Express.js a MongoDB.

Na prepojenie je teda využitá prvá spomínaná technológia. Pomocou nej sa aplikácia dopytuje na databázu, poprípade iné API, ktoré by sa v budúcnosti napojilo na tento softvér.

V databáze bude vytvorených niekoľko tzv. kolekcií. Tieto kolekcie v sebe budú udržiavať rôzne dokumenty, ktoré budú kategorizované na základe vlastného typu. Práve kolekcie sú v podstate kategórie, podľa ktorých bude aplikácia ukladať jednotlivé dáta. V prípade, že sa registruje nový užívateľ, sa budú dáta tohto nového užívateľa ukladať do kolekcie *users*. V týchto dokumentoch budú uložené dáta typu užívateľské meno, dátum vytvorenia, MAC adresa zariadenia na identifikáciu v prípade nejakej poruchy alebo zneužívania aplikácie.

Na ukladanie tiketov bude slúžiť kolekcia *tickets*. V tejto kolekcii budú uložené dokumenty, taktiež s unikátnym identifikátorom *id*, ktoré v sebe budú niesť dáta, ako názov tiketu, dátum vytvorenia, lokalita, kde bol daný záznam vytvorený, opis tiketu, autor tiketu, atď.

Týchto kolekcií je v databáze niekoľko, keďže je potrebné ešte kategorizovať samotné lokality, ktoré budú pri každom zadávaní QR kódu danej sekcie skladu stiahnuté na front-end a na základe tohto kódu bude vybratý konkrétny dokument z kolekcie *locations*. Týmto sa potom užívateľovi v aplikácii vyhodnotí jeho základná lokalita, v ktorej sa nachádza, a z ktorej môže tieto tikety posilať. V prípade manažéra by išlo o filtrovanie tiketov podľa daného QR kódu, ktorý by si na mieste mohol pomocou mobilného fotoaparátu oskenovať na svoje mobilné zariadenie, a ako správca vyčleniť všetky záznamy, ktoré sa týkajú len daného miesta.



Obr. 4.1: Rozhranie MongoDB

Databázu bude mať na starosti vývojár, ktorý na nej pracoval. Po prvotnej inštalácii a nastavení však najskôr nebude veľká potreba pre nejakú údržbu. Jedine v prípade, že ide o vytváranie umelých dát na testovanie nových funkcionalít, poprípade zmeny podmienok v aplikácii MongoDB.

Zo začiatku budú vytvorené kolekcie, do ktorých budú vytvorené dokumenty pomocou Express.js prostredníctvom softvéru Rohcket. Každý z týchto dokumentov bude samozrejme obsahovať automaticky vygenerované *id*, dosadené databázou. Tá bude obsahovať tieto kolekcie:

- **Locations.** Dokumenty tejto kolekcie zodpovedajú všetkým konkrétnym lokalitám v skladoch podniku. Na základe vygenerovaných *id* budú vytvorené QR kódy, ktoré budú tieto identifikátory šifrovať. Po následnom načítaní QR kódu sa aplikácia pokúsi dotazovať na danú lokalitu, pričom sa pokúsi stiahnuť všetky detaily o lokalite z tejto kolekcie. V prípade, že aplikácia nebude schopná lokalitu v databáze nájsť, bude užívateľovi zobrazená chyba. Tento mechanizmus bude slúžiť ako prevencia pred zadávaním neplatných lokácií (napr. pri zneužití aplikácie).
- **Tags.** Pôjde o kategórie a priority, ktoré budú užívatelia používať pri označovaní tiketov. Dokumenty budú rozlišované na základe poľa *type*, ktorý bude určovať typ daného objektu.

- **Tickets.** V tejto kolekcii sú ukladané všetky tikety, ktoré budú vytvorené prostredníctvom aplikácie. Ide o najobsiahlejší typ dokumentu zo všetkých, keďže ide v podstate o jadro celej aplikácie a dôvod, prečo aplikácia vlastne existuje.
- **Users.** Kolekcia ukladá všetkých užívateľov, či už pôjde o manažérov alebo skladníkov. V samotnom dokumente bude práve na tento účel používané pole s názvom *isManager*, ktorý bude vyhodnotený priamo vývojárom na základe požiadavky z vedenia. V prípade, že je nový registrovaný užívateľ manažér, bude toto pole zmenené na hodnotu *true*. V tomto objekte sú ďalšie pomocné polia, ako napríklad pole *macAddress*, ktoré bude využívané na prípadné zablokovanie zariadenia v prípade, že niekto bude aplikáciu zneužívať. Taktiež môže byť táto adresa využitá pri blokovaní užívateľov v rámci firemnej siete.

Presná štruktúra všetkých dokumentov je dostupná na obrázku 4.2.

Ako bolo spomenuté, na prepojenie servera s aplikáciou bol použitý framework Express.js. Ide o štandardnú konfiguráciu, ktorá je priamo napojená na server MongoDB, konkrétne server s názvom *rocketdb*, ktorý je momentálne spustený na cloude, avšak firma ho chce postupne migrovať na svoje vlastné servery, keďže chcú databázu udržiavať vo vlastnej infraštruktúre.



kolekcia
tags
dokument

```
{
  _id: ObjectId,
  description: string,
  name: string,
  type: "category" | "priority",
}
```

kolekcia
tickets
dokument

```
{
  _id: ObjectId,
  category: string,
  completedAt?: Date,
  createdAt: Date,
  createdBy: ObjectId,
  managerTicket: boolean,
  locationId: ObjectId,
  priority: 0 | 1 | 2,
  ticketDescription: string,
  ticketName: string,
  ticketStatus: "active" | "completed",
}
```

kolekcia
locations
dokument

```
{
  _id: ObjectId,
  building: string,
  city: "Brno" | "Praha",
  section: string,
}
```

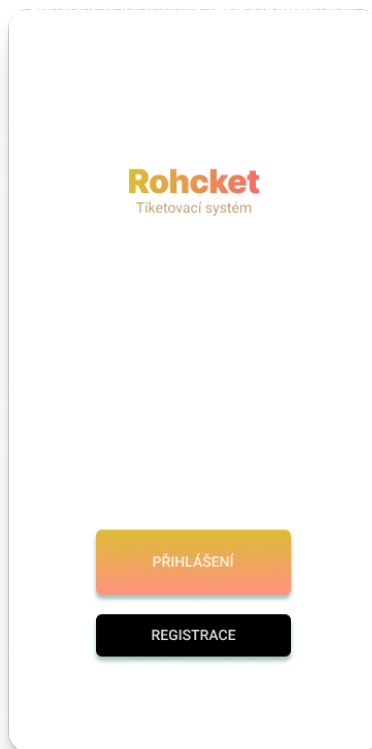
kolekcia
users
dokument

```
{
  _id: ObjectId,
  createdAt: Date,
  email: string,
  isManager: boolean,
  lastLogin: Date,
  macAddress: string,
  mainLocation: ObjectId,
  nameFirst: string,
  nameLast: string,
  ticketsCreated: string[],
}
```

Obr. 4.2: Štruktúra databázy (zdroj: vlastná tvorba)

Front-end

Po spustení aplikácie je užívateľ privítaný obrazovkou, ktorá mu dáva na výber sa zaregistrovať alebo prihlásiť. V oboch prípadoch ide o klasické formuláre, kde musí užívateľ vyplniť e-mail a heslo svojho účtu, teda zadať svoje autentifikačné údaje. Je treba podotknúť, že registrácia funguje len pre obyčajných užívateľov, pričom administrátori (teda správcovia skladu) sú registrovaní manuálne do databázy. Pôvodne sa mali vedieť registrovať aj administrátori, avšak týmto spôsobom sa predchádza zneužívaniu tohto typu registrácie. Ak má správca skladu už vytvorený účet v databáze, môže sa jednoducho prihlásiť cez tlačítko "Prihlášení".



Obr. 4.3: Úvodná obrazovka aplikácie (zdroj: vlastná tvorba)

To, ako sa užívateľ zaregistruje, či už ako administrátor alebo ako zamestnanec, rozhodne, do ktorej obrazovky bude následne presunutý.

Ak je užívateľ skladník, bude presunutý do rozhrania, v ktorom má naskenovať QR kód danej lokácie, v ktorej sa nachádza. V prípade, že sa jedná o manažérsky účet, je manažér priamo presunutý do obrazovky spravovania všetkých tiketov, kde potom môže skenovanie QR kódu využiť na ich filtrovanie.

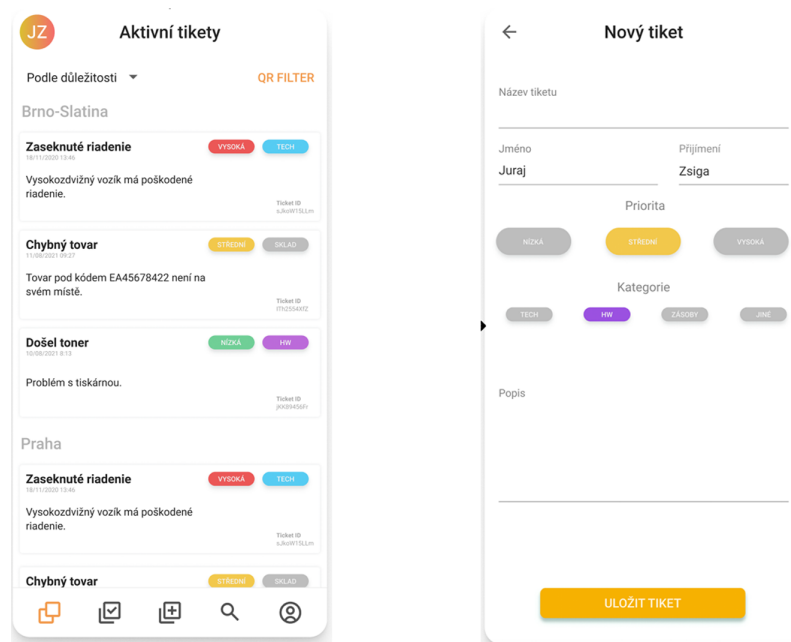


Obr. 4.4: Rozhranie pre skenovanie QR kódu (zdroj: vlastná tvorba)

Na obrázku 4.5 je možné vidieť, že správcovia majú prístup k piatim rôznym rozhraniam, pričom zamestnanci s klasickými právami budú mať prístup len k vytváraniu tiketov, k svojim tiketom a k úprave svojho profilu.

Opis daných možností (zľava dolného menu):

- **Aktívne tikety.** Ide o záznamy, ktoré ešte vyriešené neboli a samotný obsah týchto tiketov je stále aktuálny. Pomocou tohto rozhrania má manažér k dispozícii všetky tikety vo všetkých lokalitách, pričom si ich vie filtrovať podľa svojej potreby. Front-end je nastavený tak, že je možné manažérovi obmedziť záznamy len na jednu lokalitu.
- **Dokončené tikety.** Na tejto obrazovke môže správca vidieť všetky dokončené karty z vybraných lokalít. Tento „screen“ sa môže zísť v prípade, že treba kartu recenzovať, poprípade vrátiť späť medzi aktívne karty, ak by teda vypracovanie danej úlohy bolo neadekvátne.
- **Pridať tiket.** V tomto rozhraní bude manažér vedieť vytvárať tikety. Užívateľ bude musieť zadať názov tiketu, popis, poprípade bude môcť pridať fotky alebo video. Tento nový záznam bude následne uložený do databázy do kolekcie *tickets* ako nový dokument s unikátnym identifikátorom.
- **Hľadať tiket.** Ak správca bude potrebovať rýchlo nájsť tiket, či už pôjde o dokončený alebo aktívny záznam, môže využiť túto funkcionality, pomocou ktorej dokáže nahliadnuť do daného záznamu a jeho detailov. Funkcionalita by fungovala na základe dopytovania sa na databázu pomocou jedného z týchto argumentov: názov tiketu, kľúčové slovo (tag), dátum vytvorenia, autor tiketu, atď.
- **Profil.** V poslednej karte si manažér môže meniť svoje nastavenia, ako je heslo, e-mail a lokalita. Tento screen bude využívaný skôr zriedkavo, pričom najväčšie využitie bude mať samotná zmena lokality.

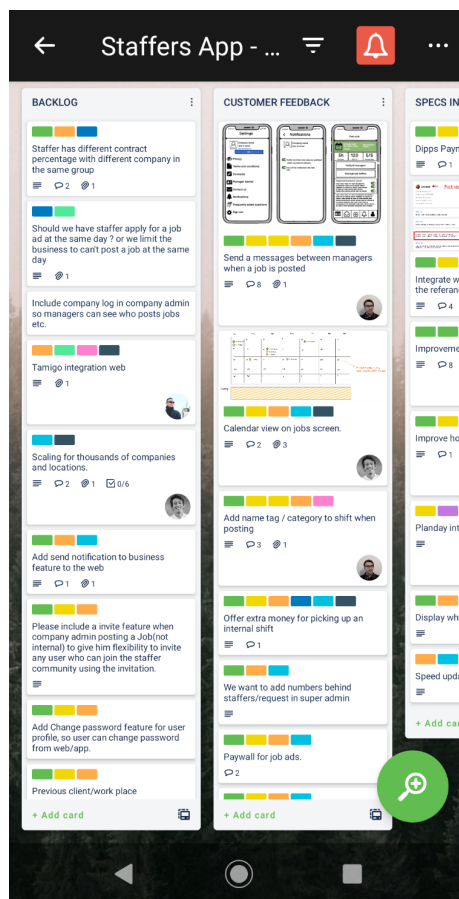


Obr. 4.5: Rozhranie pre administrátorov a vytváranie tiketu

Z tohto výberu bude mať bežný užívateľ prístup len k vytvoreniu tiketu 4.5 (vpravo). Týmto chce firma urobiť softvér čo najjednoduchší pre použitie radovými zamestnancami v sklade. Samotná integrácia a ovládanie aplikácie bude z tohto hľadiska veľmi jednoduché a tým pádom nebude potrebné vynaložiť toľko zdrojov na školenie zamestnancov.

4.8 Porovnanie navrhnutého riešenia so súčasným

V tejto podkapitole bude vykonané porovnanie súčasnej aplikácie a navrhovanej aplikácie. Vzhľadom na to, že Trello je externá aplikácia, nie sú uvádzané informácie typu: akú používa Trello databázu, aký jazyk používa na back-end, atď.



Obr. 4.6: Rozhranie Trello (zdroj: screenshot Android aplikácie)

Na obrázku 4.5 je možné vidieť návrh aplikácie pre Rohlík, pričom na obrázku 4.6 je zobrazené rozhranie aplikácie Trello. Na prvý pohľad vyzerajú aplikácie pomerne odlišne, ale spĺňajú podobné funkcie. Čo sa týka funkcionalít z hľadiska vytvárania tiketov a priradenia zamestnancov, obe riešenia ich ponúkajú.

Hlavným dôvodom pre zmenu, resp. naprogramovanie vlastnej aplikácie pre podnik, je rýchly rozvoj skladových priestorov. Dôvodom je zvýšený dopyt v dôsledku prísnejších opatrení.

Ďalším dôvodom je prístup k dátam, teda prístup k databáze a jej spravovaniu. Keďže Trello je externá aplikácia, neponúka svojim užívateľom žiaden prístup k svojej databáze vo svojej verzii zadarmo. V prípade vlastnej aplikácie by analyzovaný podnik mohol sám spravovať vlastnú databázu, môže určiť stavbu dokumentov a kolekcií, rozhodnúť či bude relačná, atď.

Veľkým rozdielom je taktiež náročnosť aplikácie Trello na hardvér mobilných zariadení. Tento softvér má v sebe zabudovaných pomerne veľa animácií, čím chce najskôr osloviť možno aj mladšie generácie, poprípade spríjemniť užívateľovi službu. Nuž, pri horších zariadeniach môže byť tento softvér práve kvôli tomuto veľmi pomalý, čiže skladníkom môže zabrať dosť času pracovať s kartami v Trelle.

Čo sa týka školenia ohľadom nového softvéru, bude stačiť jednoduchá verzia, poprípade pár návodov vo forme videa, ktoré si zamestnanci budú môcť pozrieť vo voľnom čase, poprípade pri nástupe do práce. Obe aplikácie majú jednoduchý návrh základného používania, pričom tie náročnejšie funkcie budú využívané správcami skladu, poprípade inými zamestnancami, ktorí budú karty manažovať prostredníctvom portálu.

Vlastnosti	Trello	Rohcket
Identifikácia lokality	Nie	Áno
Vytváranie tiketov	Áno	Áno
Prístup k databáze	Nie	Áno
Česká lokalizácia	Áno	Áno
Anglická lokalizácia	Áno	Nie
Mesačný poplatok	Áno	Nie
Lokálny „service desk“	Nie	Áno
Flexibilita aplikácie (doprogramovanie nových funkcionalít)	Nie	Áno
Prístup len v rámci firemnej siete	Nie	Áno

Tabuľka 4.12: Porovnanie aplikácií Trello a Rohcket

4.9 Analýza nákladov

Podľa výslednej časovej analýzy je možné zhruba určiť celkové náklady na implementáciu softvéru do firemného prostredia Rohlíku. Pri časovom odhade a celkovej práci potrebnej na vyvinutie tejto aplikácie sa celková cena vývoja aplikácie mala vyšplhať zhruba na 100 000 Kč (čiže cca 25 000 Kč na sprint). Treba podotknúť, že v tomto prípade sa neúčtuje hodinová sadzba, ale celková cena za projekt, ktorý však bude naďalej udržiavaný, čo vyžaduje uvoľnenie ďalších finančných prostriedkov aj po oficiálnom spustení plnej verzie mobilnej aplikácie. Pri celkovom zhodnotení projektu sa teda dospelo k záveru, že vývoj softvéru bude naďalej udržiavaný jeho tvorcom pri cene 50 000 Kč/mesiac. Vzhľadom na jednoduchosť aplikácie podnik zatiaľ neuvažuje o rozšírení vývojového tímu starajúceho sa o aplikáciu. Ďalšou položkou je licencia na účet iOS, prostredníctvom ktorého je možné distribuovať aplikáciu súkromne na iOS zariadenia. Ide o cenu 6 400 Kč/rok, teda 533 Kč/mesiac.

Vývoj	
Položka	Cena (Kč)
Softvér	100 000
iOS licencia	533 / mesiac
Udržiavanie aplikácie	50 000 / mesiac

Tabuľka 4.13: Náklady na vývoj aplikácie

Cena za hardvér, ako je server, prístupové body WiFi v skladoch a ich inštaláciu, prípadne služobné mobilné zariadenia, je možné oceniť zhruba na 195 000 Kč.

Konečná cena je teda 200 000 Kč za počiatočné spustenie, vrátane inštalácie hardvéru a 50 000 Kč za každý ďalší mesiac. Zatiaľ však pôjde len o Brnenskú pobočku, v prípade rozšírenia do ďalších miest bude potrebný nákup nového hardvéru, čím sa cena znovu zvýši.

Pri zvýšenom objeme zákazníkov a ich objednávok, by sa táto investícia mala za nejakú dobu vrátiť, keďže navrhovaná zmena pomôže firme oveľa rýchlejšie zaznamenať a dohľadať na problémy, ktoré sa vyskytujú v ich skladových priestoroch. Je bežné, že malé nedostatky,

Hardvér	
Položka	Cena (Kč)
Prístupové body WiFi + kabeláž	50 000
Náhradné mobilné zariadenia	50 000
Server	50 000
Inštalácia	45 000

Celkovo	195 000
----------------	---------

Tabuľka 4.14: Náklady na hardvér a jeho inštaláciu

napr. pokazená skladovacia technika, potraviny po záruke, atď., sa vyskytnú v sklade denne. Ak budú tieto chyby rýchlejšie odstránené, môže firma zlepšiť svoju reputáciu na trhu ešte viac, keďže celý proces dodávania potravín pre zákazníka začína práve v sklade, kde dodanie jednej z položiek môže byť pre sklad práve problematické, čo stojí nejaký čas, a čo v konečnom dôsledku môže firmu stáť aj zákazníka.

Celkové náklady	
Položka	Cena (Kč)
Jednorázové platby	295 000
Mesačné platby	50 550 / mesiac

Tabuľka 4.15: Výsledné náklady

4.10 Prínos nového riešenia

Vybraná skupina skladníkov si aplikáciu pochvaľuje a dodáva konštantnú spätnú väzbu. Páči sa im hlavne jednoduchšie rozhranie, ktoré je aj rýchlejšie na ovládanie, vďaka jednoduchším animáciám, ako v aplikácii Trello.

Už po prvých dňoch využívania sa vedenie technického oddelenia zhodlo na tom, že aplikáciu by mal taktiež sprevádzať webový portál pre administrátorov, ktorí by mali vedieť karty vytvárať a spravovať prostredníctvom internetového prehliadača. Odhliadnuc od toho si správcovia pochvaľujú filter kariet, vzhľad aplikácie a jej rýchlosť.

Vzhľadom na krátku dobu od implementácie ešte nie je možné určiť finančný prínos tejto investície, avšak pri ušetrenom čase v porovnaní s aplikáciou Trello firma odhaduje, že by sa im investícia mala vrátiť zhruba do dvoch rokov. Treba totiž brať do úvahy, že distribúcia aplikácie môže vďaka súčasným podmienkam trvať dlhšie, ako sa zdalo.

Záver

V prvej kapitole boli uvedené ciele práce, ktorými bolo navrhnuť, vytvoriť a implementovať softvérovú aplikáciu do firemného prostredia na základe analýz, ktoré odhalili nedostatok v rámci podniku a jeho procesov. Firma, do ktorej bola aplikácia implementovaná, je Veľká Pecka s.r.o., známa pod názvom Rohlík.cz, ktorá operuje najmä ako online supermarket.

Druhá kapitola opísala teoretické východiská práce, kde išlo o rôzne technológie pri programovaní aplikácie, alebo o projektové metódy, ktoré boli využité v jednotlivých analýzach firmy. Táto časť slúžila najmä na priblíženie danej problematiky zo všeobecného hľadiska, pričom mala vysvetliť základné pojmy čitateľovi, ktorý sa s nimi predtým nikdy nestretol.

V ďalšej kapitole bola v skratke predstavená firma a následne už boli dané metódy z druhej kapitoly využité pri analýze firmy. Vonkajšie prostredie firmy bolo analyzované ako prvé metódou SLEPT, za čím nasledovala analýza konkurencie firmy a potenciálnych hrozieb. Kapitola taktiež opisovala firmu a jej praktiky pomocou metódy 7S. Na záver bola využitá analýza SWOT.

Na základe analýz bolo zistené, že je potrebné vylepšiť skladovú infraštruktúru mobilnou aplikáciou, ktorá by pomohla zlepšiť komunikáciu medzi zamestnancami v rámci skladu. Preto bol v štvrtej kapitole opísaný návrh takéhoto softvéru s názvom Rohcket. Táto časť práce zahŕňala porovnanie softvéru Trello, ktorý firma používa, s navrhovanou aplikáciou. Projektové analýzy zahŕňali Lewinov model, metódu RIPRAN a časový plán činností pomocou agilných metód. V krátkosti boli taktiež uvedené náklady. Záver kapitoly ponúka prehľad nad vlastným návrhom riešenia, jeho zavedením a jeho prínosom.

Keďže bola situácia firmy analyzovaná a aplikácia úspešne navrhnutá, vytvorená a implementovaná do firemného prostredia, môžu sa ciele tejto práce, teda aj samotná práca, považovať za úspešné.

Literatúra

- [1] *Welcome to the MongoDB Documentation* [online]. MongoDB, Inc [cit. 11-02-2021].
Dostupné z: <https://docs.mongodb.com/>.
- [2] *McKinsey 7S* [online]. ManagementMania, júl 2015 [cit. 17-02-2021]. Dostupné z:
<https://managementmania.com/cs/mckinsey-7s>.
- [3] *Malí výrobci budou moct prodávat na Rohlik.cz. Kvalitu ohlídá Sklizeno* [online].
Zboží a Prodej, marec 2018 [cit. 17-02-2021]. Dostupné z:
<https://www.zboziaprodej.cz/2018/03/20/mali-vyrobc-budou-moct-prodav-at-na-rohlik-cz-kvalitu-ohlida-sklizeno/>.
- [4] *SLEPT Analysis* [online]. MBA Skool, august 2018 [cit. 19-11-2020]. Dostupné z:
<https://www.mbaskool.com/business-concepts/marketing-and-strategy-terms/8377-slept-analysis.html>.
- [5] *Bezpečnostní opatření - Rohlík.cz* [online]. Rohlík.cz, marec 2020 [cit. 17-02-2021].
Dostupné z: https://www.rohlik.cz/tema/bezpecnostni-opatreni?_ref=hp.
- [6] *Introduction to Expo* [online]. Expo, november 2020 [cit. 17-11-2020]. Dostupné z:
<https://docs.expo.io/>.
- [7] *Understanding client-side JavaScript frameworks* [online]. Mozilla, august 2020 [cit. 09-11-2020]. Dostupné z: https://developer.mozilla.org/en-US/docs/Learn/Tools_and_testing/Client-side_JavaScript_frameworks.
- [8] *Velká Pecka s.r.o. - Výroční zpráva za účetní období 01.05.2019 - 30.04.2020* [online].
Justice.cz, apríl 2020 [cit. 17-02-2021]. Dostupné z:
<https://or.justice.cz/ias/ui/vypis-sl-detail?dokument=63565817&subjektId=867007&spis=304197>.

- [9] *Život v Rohlíku* [online]. Rohlík.cz, august 2020 [cit. 11-02-2021]. Dostupné z: <https://www.zivotvrohliku.cz/>.
- [10] *ECCCSA 2020 Winners* [online]. ECCCSA, marec 2021 [cit. 5-4-2021]. Dostupné z: <https://www.ecccsa.com/2020-winners/>.
- [11] *Rohlík.cz (Android)* [online]. Google LLC, február 2021 [cit. 17-02-2021]. Dostupné z: <https://play.google.com/store/apps/details?id=cz.rohlik.app&hl=en&gl=US>.
- [12] *Rohlík.cz (iOS)* [online]. Apple Inc., február 2021 [cit. 17-02-2021]. Dostupné z: <https://apps.apple.com/cz/app/rohlik-cz/id975560575>.
- [13] *TIOBE Index for March 2021* [online]. TIOBE, marec 2021 [cit. 03-25-2020]. Dostupné z: <https://www.tiobe.com/tiobe-index/>.
- [14] *Česká Republika: Politická a ekonomická situace* [online]. EACEA, február 2021 [cit. 17-02-2021]. Dostupné z: https://eacea.ec.europa.eu/national-policies/eurydice/content/political-and-economic-situation-21_cs.
- [15] CHRISTENSEN, P. *Backend Definition* [online]. Sharpened Productions, apríl 2020 [cit. 02-25-2020]. Dostupné z: <https://techterms.com/definition/backend>.
- [16] CORONEL, C., MORRIS, S. a ROB, P. *Database Systems: Design, Implementation, and Management*. 9. vyd. Cengage Learning, 2011. ISBN 978-0-538-74884-1.
- [17] DHADUK, H. *Node.JS Use Case: When How Node.JS Should be Used* [online]. EACEA, október 2019 [cit. 17-02-2021]. Dostupné z: <https://www.simform.com/nodejs-use-case/>.
- [18] GODBOLT, M. *Frontend Architecture for Design Systems*. O'Reilly Media, 2016. ISBN 978-1-491-92678-9.
- [19] GÁLA, L., POUR, J. a ŠEDIVÁ, Z. *Podniková informatika*. 2. vyd. Grada Publishing, 2009. ISBN 978-80-247-2615-1.
- [20] HANZELKOVÁ, A., KEŘKOVSKÝ, M., MATHAUSER, M. a VALSA, O. *Business strategie: Krok za krokem*. C.H. Beck, 2013. ISBN 978-80-7400-455-1.
- [21] HAVERBEKE, M. *Eloquent JavaScript: A Modern Introduction to Programming*. 2. vyd. No Starch Press, 2015. ISBN 978-1-59327-584-6.

- [22] HAVLÍK, A. *Velké srovnání českých online supermarketů* [online]. Refresher, apríl 2018 [cit. 17-02-2021]. Dostupné z: <https://refresher.sk/53759-Velke-srovnani-ceskych-online-supermarketu-U-koho-se-vyplati-nakupovat-jak-resi-problemy-s-nakupem-a-kde-maji-nejvetsi-vyber-potravin>.
- [23] KOCOURKOVÁ, M. *Kde nakupovat potraviny online? Srovnání prodejců!* [online]. VímVíc, marec 2020 [cit. 17-02-2021]. Dostupné z: <https://www.vimvic.cz/clanek/kde-nakupovat-potraviny-online-srovnani-prodejcu>.
- [24] KUBÁTOVÁ, Z. *Rohlik.cz rychleji rozšíří rozvoz na venkov. Zaměstná až 700 lidí* [online]. Seznam Zprávy, marec 2020 [cit. 17-02-2021]. Dostupné z: <https://www.seznamzpravy.cz/clanek/rohlik-cz-rychleji-rozsiri-rozvoz-na-venkov-zamestna-az-700-lidi-95764>.
- [25] LACKO, B. *RIPRAN* [online]. RIPRAN, august 2020 [cit. 6-4-2021]. Dostupné z: <https://ripran.cz/>.
- [26] MISTRY, J. *10 Popular Web Frameworks for Web App Development in 2020*. Monocubed, november 2020 [cit. 1-12-2020]. Dostupné z: <https://www.monocubed.com/10-most-popular-web-frameworks/>.
- [27] OCCHINO, T. *React Native: Bringing modern web techniques to mobile* [online]. Facebook, marec 2015 [cit. 11-11-2020]. Dostupné z: <https://engineering.fb.com/2015/03/26/android/react-native-bringing-modern-web-techniques-to-mobile/>.
- [28] PORTER, M. E. *Konkurenční strategie: Metody pro analýzu odvětví a konkurentů*. Victoria Publishing, 1994. ISBN 80-85605-11-2.
- [29] RAIS, K. *Risk management: Studijní text pro kombinovanou formu studia*. Akademické nakladatelství CERM, 2007. ISBN 978-80-214-3510-0.
- [30] RAIS, K. a SMEJKAL, V. *Řízení rizik ve firmách a jiných organizacích*. 3. vyd. Grada Publishing, 2010. ISBN 978-80-247-3051-6.
- [31] RIEHLE, D. *Framework Design: A Role Modeling Approach*. Zürich, SUI, 2000. Ph.D. Thesis. ETH Zürich. Dostupné z: <https://riehle.org/computer-science/research/dissertation/diss-a4.pdf>.

- [32] RUBIN, K. S. *Essential Scrum*. Addison-Wesley, 2012. ISBN 978-0-13-704329-3.
- [33] SCHWALBE, K. *Řízení projektů v IT: Kompletní průvodce*. Computer Press, 2011. ISBN 978-80-251-2882-4.
- [34] SUBRAMANIAN, V. *Pro MERN Stack*. 2. vyd. Apress, 2019. ISBN 978-1-4842-4391-6.
- [35] ČTK. *V Globusu nakoupíte z domova. Řetězec spustil vlastní e-shop* [online]. Aktuálně.cz, október 2018 [cit. 17-02-2021]. Dostupné z:
<https://zpravy.aktualne.cz/ekonomika/na-internet-miri-dalsi-z-velkych-obchodnich-retezcu-globus/r~830c7a98d2d711e89271ac1f6b220ee8/>.

Zoznam obrázkov

2.1	Základné skupiny prostriedkov tvoriace ICT [19]	15
2.2	Príklad štruktúry záznamu (zdroj: vlastná tvorba)	18
2.3	Dynamicky zobrazené tikety v aplikácii (zdroj: vlastná tvorba)	19
2.4	Rozhranie aplikácie Figma	21
2.5	Využitie Node.js v praxi [17]	24
2.6	Rozdiel medzi databázami MongoDB a SQL [1]	26
2.7	Rámec 7S od firmy McKinsley [2]	30
2.8	Fázy metódy RIPRAN [25]	34
3.1	Logo Rohlík.cz [9]	37
4.1	Rozhranie MongoDB	69
4.2	Štruktúra databázy (zdroj: vlastná tvorba)	71
4.3	Úvodná obrazovka aplikácie (zdroj: vlastná tvorba)	72
4.4	Rozhranie pre skenovanie QR kódu (zdroj: vlastná tvorba)	73
4.5	Rozhranie pre administrátorov a vytváranie tiketu	75
4.6	Rozhranie Trello (zdroj: screenshot Android aplikácie)	76

Zoznam tabuliek

3.1	Tabuľka SWOT (zdroj: vlastná tvorba)	51
4.1	Tabuľka síl pre Lewinov model (zdroj: vlastná tvorba)	56
4.2	Tabuľka stupňov pravdepodobnosti [25]	59
4.3	Tabuľka ohodnotenia dopadu	60
4.4	Tabuľka výsledných možných hodnôt rizika [25]	60
4.5	Tabuľka rizík pomocou metódy RIPRAN	61
4.6	Tabuľka rizík po zavedení opatrení	62
4.7	Porovnanie odhadovaných a reálnych počtov MD v sprintoch	63
4.8	Sprint 1 - tabuľka činností	64
4.9	Sprint 2 - tabuľka činností	65
4.10	Sprint 3 - tabuľka činností	66
4.11	Sprint 4 - tabuľka činností	67
4.12	Porovnanie aplikácií Trello a Rohcket	77
4.13	Náklady na vývoj aplikácie	78
4.14	Náklady na hardvér a jeho inštaláciu	79
4.15	Výsledné náklady	79